(2)

SIMULATION OF OIL SLICK TRANSPORT IN GREAT LAKES CONNECTING CHANNELS

Volume II: User's Manual for the River Oil Spill Simulation Model

H.T. Shen
P.D. Yapa
M.E. Petroski

Report No. 86-2
March 1986
Department of Civil and Environmental Engineering
Clarkson University
Potsdam · New York · 13676

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) Report No. 86-2	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Clarkson University 6b. OFFICE SYMBOL *(If applicable)*	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS *(City, State, and ZIP Code)* Department of Civil and Environmental Engineering Potsdam, NY 13676	7b. ADDRESS *(City, State, and ZIP Code)*
8a. NAME OF FUNDING/SPONSORING ORGANIZATION U.S. Army Corps of Engineers 8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DACA 33-85-C-0001
8c. ADDRESS *(City, State, and ZIP Code)* Detroit District P.O. Box 1027 Detroit, MI 48231	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*
Simulation of Oil Slick Transport on Great Lakes Connecting Channels; Volume II: User's Manual for the River Oil Spill Simulation Model (ROSS)

12. PERSONAL AUTHOR(S)
Shen, H.T., P.D. Yapa, and M.E. Petroski

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
Final	FROM ____ TO ____	March 1986	229

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUB-GROUP	ROSS, oil slick, grid boxes, ice regions velocity distributions, plot of slick locations

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*
In this study, two computer models named as ROSS and LROSS are developed for simulating oil slick transport in rivers and lakes, respectively. The oil slick transformation processes considered in these models include advection, spreading, evaporation and dissolution. These models can be used for slicks of any shape originated from instantaneous or continuous spills in rivers and lakes with or without ice covers. Although developed for the need of the connecting channels in the upper Great Lakes, including the Detroit River, Lake St. Clair, St. Clair River, and St. Marys River, these models are site independent and can be used to other rivers and lakes. The programs are written in FORTRAN programming language to be compatible with FORTRAN77 compiler. The models are designed to be used on both mainframe and microcomputers.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Jimmie L. Glover	22b. TELEPHONE *(Include Area Code)* (313) 226-7590 — 22c. OFFICE SYMBOL CENCE-PD-EA

DD Form 1473, JUN 86 *Previous editions are obsolete.* SECURITY CLASSIFICATION OF THIS PAGE

Unclassified

SIMULATION OF OIL SLICK TRANSPORT IN GREAT LAKES CONNECTING CHANNELS

Volume II: User's Manual for the River Oil Spill Simulation Model (ROSS)

by

Hung Tao Shen, Poojitha D. Yapa, and Mark E. Petroski

Report No. 86-2

Department of Civil and Environmental Engineering
Clarkson University
Potsdam, New York 13676

March 1986

Sponsored by

U.S. Army Corps of Engineers
Contract No. DACA33-85-C-0001

This program is furnished by the Government and is accepted and used by the recipient upon the express understanding that the United States Government makes no warranties, express or completeness, reliability, usability, or suitability for any particular purpose of the information and data contained in this program or furnished in connection therewith, and the United States Government shall be under no liability whatsoever to any person by reason of any use made thereof.

The program herein belongs to the Government. Therefore, the recipient further agrees not to assert any proprietary rights therein or to represent this program to anyone as other than a Government program.

PREFACE

The growing concern over the possible impacts of oil spills on aquatic environments has led to the development of a large number of computer models for simulating the transport and spreading of oil slicks in surface water bodies. Almost all of these models were developed for coastal environments. With the increase in inland navigation activities, oil slick simulation models for rivers and lakes are needed.

In this study, two computer models named as ROSS and LROSS are developed for simulating oil slick transport in rivers and lakes, respectivley. The study was originated by the Detroit District, U.S. Army Corps of Engineers in relation to the Great Lakes limited navigation season extension study. The oil slick transformation processes considered in these models include advection, spreading, evaporation and dissolution. These models can be used for slicks of any shape originated from instantaneous or continuous spills in rivers and lakes with or without ice covers. Although developed for the need of the connecting channels in the upper Great Lakes, including the Detroit River, Lake St. Clair, St. Clair River, and St. Mary's River, these models are site independent and can be used to other rivers and lakes.

The programs are written in FORTRAN programming language to be compatible with FORTRAN77 compiler. In addition, a user-friendly, menu driven program with graphics capability is developed for the IBM-PC AT computer, so that these models can be easily used to assist the oil spill cleanup action in the connecting channels should a spill occur.

This report series is organized in four volumes, to provide a complete description of the analytical formulation of the models, the logic and structures of the computer programs, and the instructions for using the

models. The title of these volumes are:

Volume I:	Theory and Model Formulation
Volume II:	User's Manual for the River Oil Spill Simulation Model (ROSS)
Volume III:	User's Manual for the Lake-River Oil Spill Simulation Model (LROSS)
Volume IV:	User's Manual for the Microcomputer-Based Interactive Program

ACKNOWLEDGEMENTS

This study was supported by the U.S. Army Corps of Engineers under Contract No. DACA33-85-C-0001. Steve F. Daly and Mike Ferrick of the U.S. Army Cold Regions Research and Enginering Laboratory are the contracting officer's technical representatives. The writers would like to thank both of them as well as Dan Thompson and Don Williams of the Detroit District, U.S. Army Corps of Engineers, for their cooperation and assistance throughout the study period.

The writers would also like to acknowledge the assistance provided by the following individuals during various stages of this study: J.R. Weiser and R. Thomas, Detroit District, U.S. Army Corps of Engineers; F.H. Quinn and D.J. Schwab, Great Lakes Environmental Research Laboratory, NOAA; J. Galt and T. Kaiser, NOAA; M. Sydor, Inland Water Directorage, Canada; G. Tsang and R.O. Ramseir, Environment Canada; D. Mackay, University of Toronto; J.A. McCorquodale, University of Windsor; and S. Venkatesh, Atmospheric Environment Service, Canada.

TABLE OF CONTENTS

VOLUME II. USER'S MANUAL FOR THE RIVER OIL SPILL SIMULATION MODEL (ROSS)

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The computer model ROSS for simulating the spreading of an oil slick in rivers is presented. The model first constructs the two-dimensional, depth averaged velocity distribution for the river and subsequently simulates an oil spill as identified by user supplied input. Analytical formulations used in the computer model for describing the physical processes affecting the oil and its transport are presented in Volume I. The flow chart presented in Fig. 1 outlines the structure of the computer model. Brief discussions on the computer logic and techniques will be given in this chapter. Detailed discussions on the implementation of the analytical formulation code are presented in later chapters along with input and output instructions. The model described in this manual is developed for the Great Lakes Connecting Channels, including the St. Clair River, the Detroit River, and the St. Mary's River. However, the model is programmed in a genearl manner such that it can be utilized on other rivers as well. A thorough description of the model input and output is given to aid the users in the adaptation of the model to rivers other than those which are presented. The program is also designed so that the refinement of the model elements and expansion of the model to include addition of physical and chemical processes can easily be made.

I.1 Model Implementation

Oil Slick Representation

The oil slick is represented by a user specified number of particles up to 1000. Each particle can be considered as a parcel of oil which represents an equal fraction of the entire oil slick volume. If, for

example, the user chooses 500 particles for a 50,000 barrel spill, each particle would represent 100 barrels of oil (1 barrel = 55 U.S. gal.). Treatment of the instantaneous or continuous spill cases differ only in the manner in which the particles are released and will be further explained in a later section. Whether a given spill is to be treated as instantaneous or continuous is decided automatically by the model based on spill duration and the time step to be used.

Velocity Distribution

Analysis of the transport of oil spill particles in a river requires a well defined water velocity distribution. The water velocity is eventually combined with the wind velocity to advect the oil (as will be discussed in the section on advection). To calculate these water velocities, various information of the river need to be considered. These include the total discharge, river stages, cross section geometries, river shoreline characteristics, ice conditions, and the existence of islands.

Computing the two dimensional velocity distribution in the river is done in two stages. First, a one dimensional unsteady flow model is used to generate discharge and river stages at specified cross sections referred to as nodes. The oil spill model utilizes this generated information to compute two-dimensional velocity distribution. In the oil spill model, branches are used to describe the channel configuration. The beginning and ending of these branches are selected to coincide with nodes of the one-dimensional model in most cases. In cases where it is necessary to have more branches in the oil spill model than in unsteady flow model the water levels at intermediate points are obtained through interpolation.

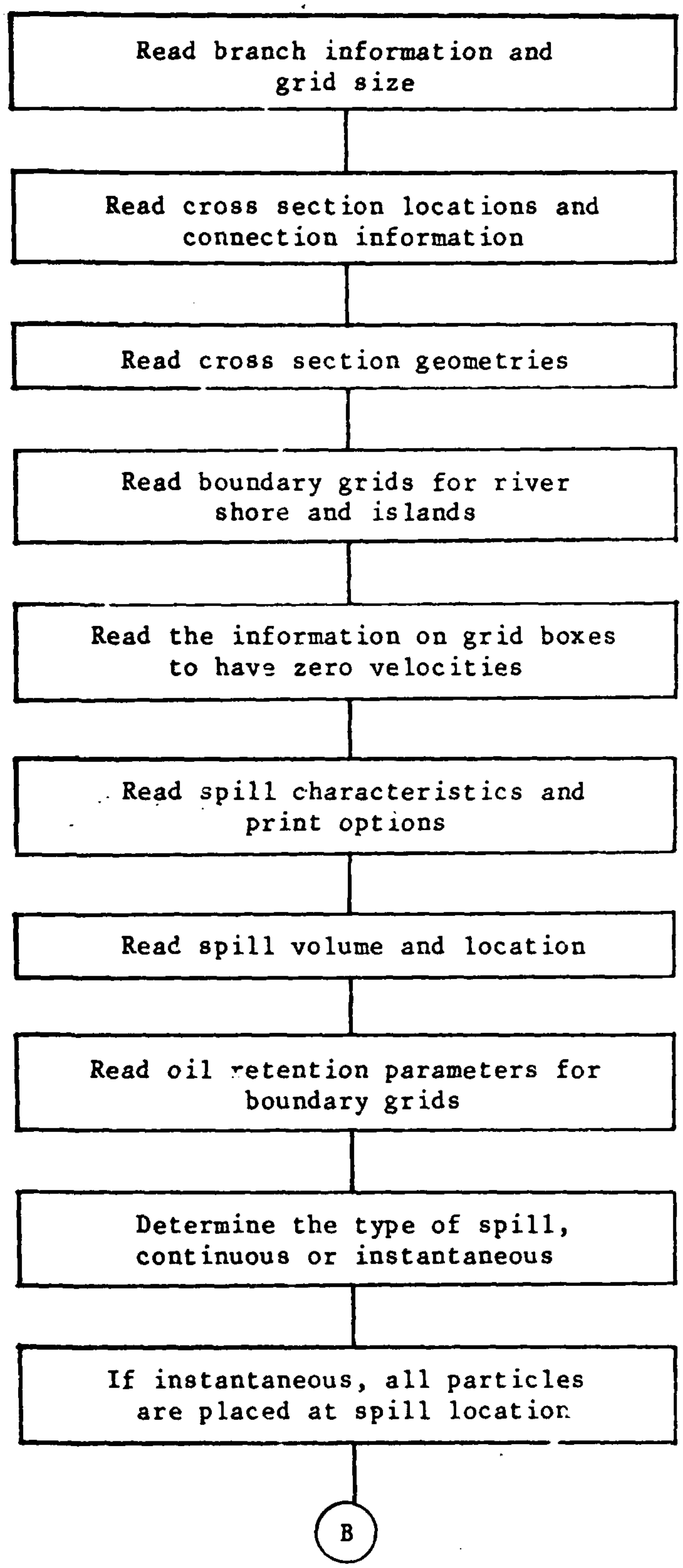


Figure 1. Block Diagram of Computer Model ROSS

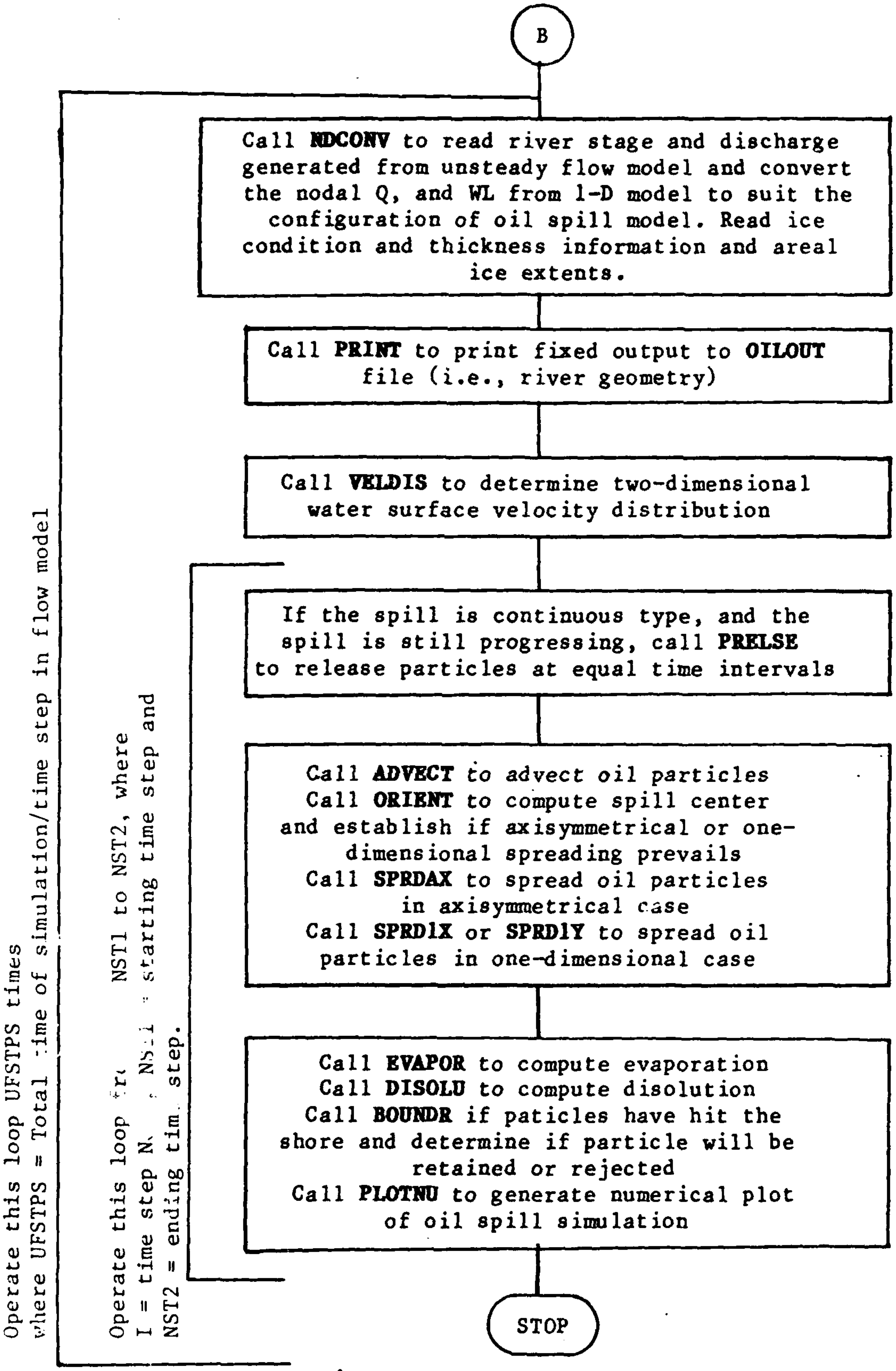
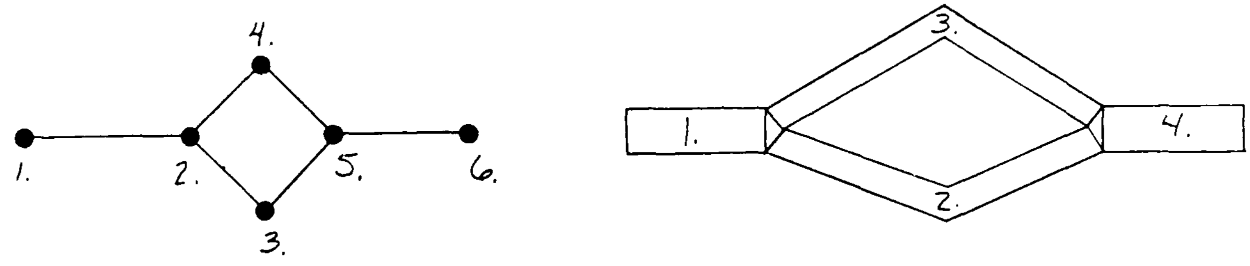


Figure 1. Block Diagram of Computer Model ROSS

Figures 2 through 6 show one-dimensional schematizations for the connecting channels of the Great Lakes.

In the event an island is encountered, it becomes necessary to divide the discharge accordingly (Fig. 2). One or more branches extend around each side of the island. Each of these branches include a portion of the total number of streamtubes in the river. No new streamtubes are added (i.e. the total number remains the same but is split bettween the upper and lower island branches according to the ratio of flow split). After the magnitude of velocities and their center point locations are computed, the direction of velocity can be obtained by connecting corresponding streamtube boundaries in two successive cross sections. The information available at this stage is the magnitude and direction (or alternatively x and y components) of velocities of the specified streamtubes at the cross sections.



a) b)

Fig. 2 Techniques for representing a river when islands are present; a) nodes b) branches

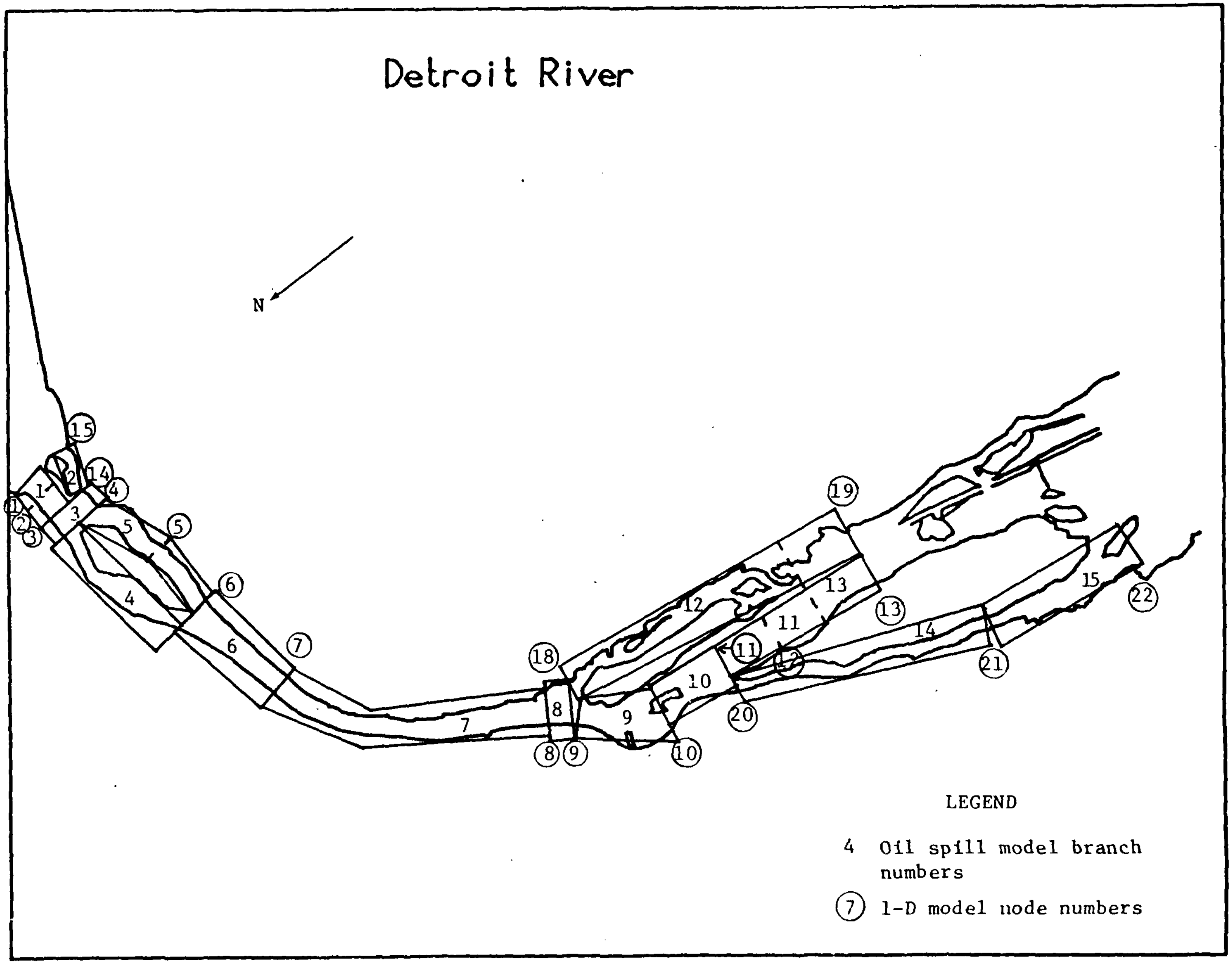


Fig. 3 Branch Configuration for Detroit River

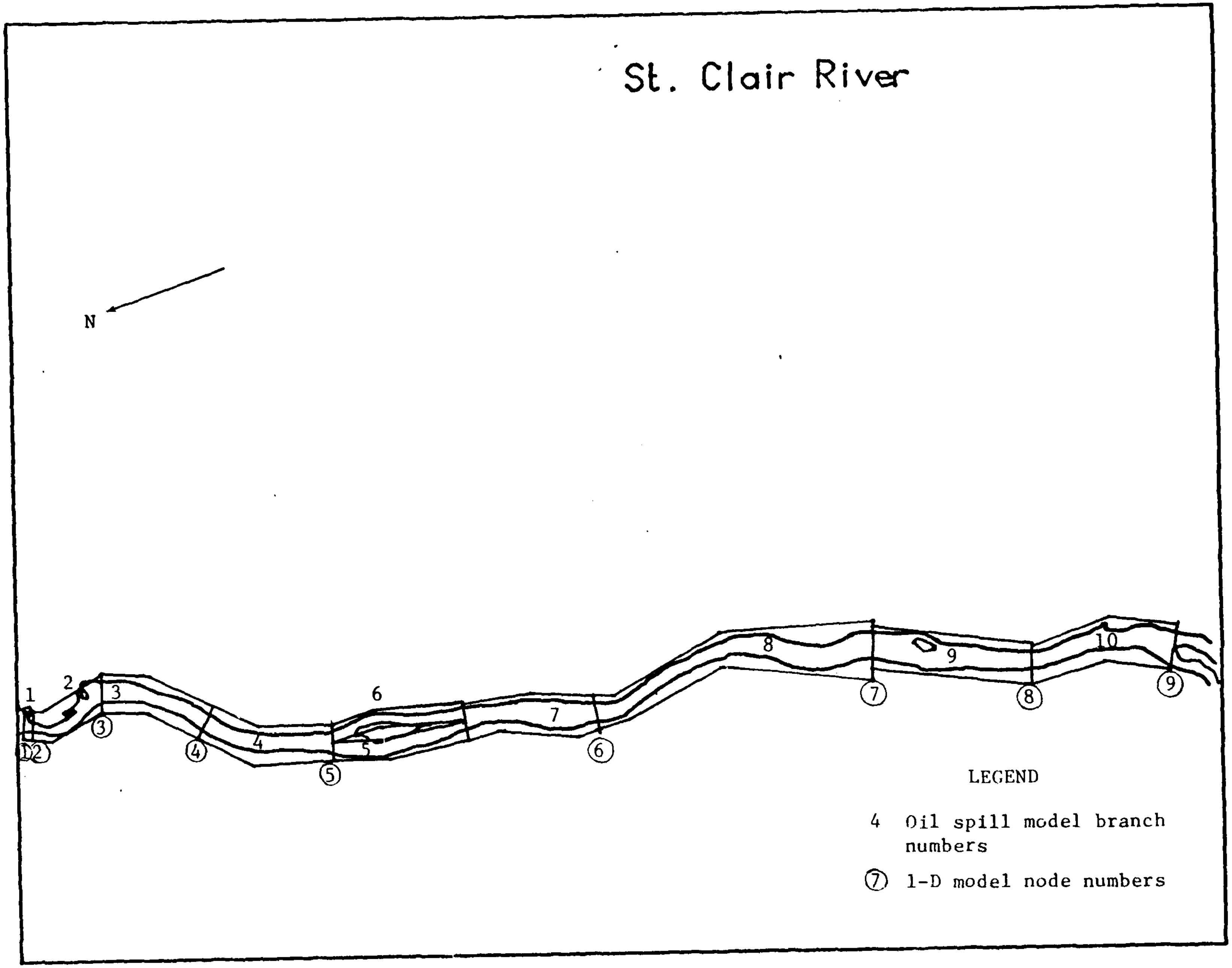


Fig. 4 Branch Configuration for St. Clair River

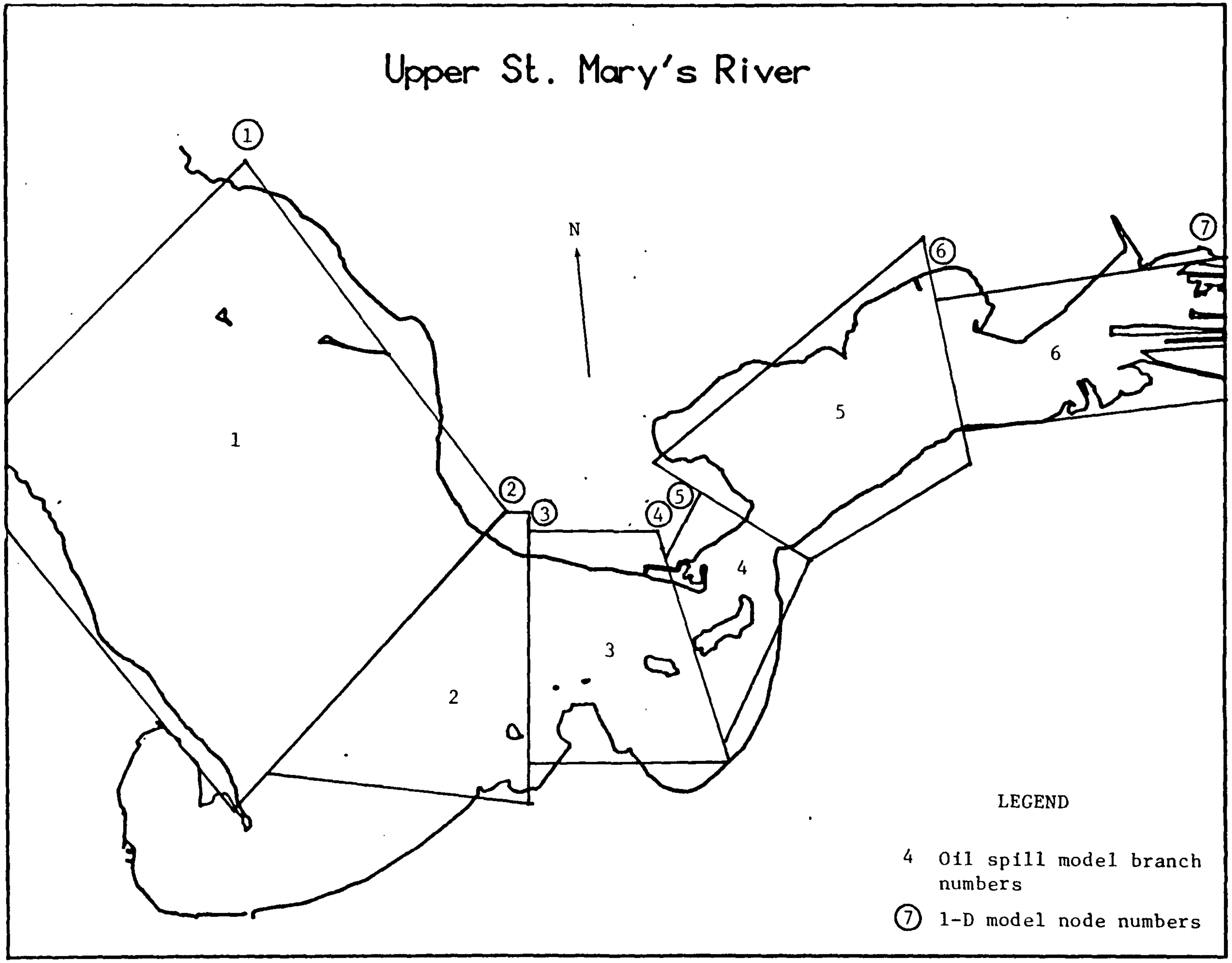


Fig. 5 Branch Configuration for Upper St. Mary's River

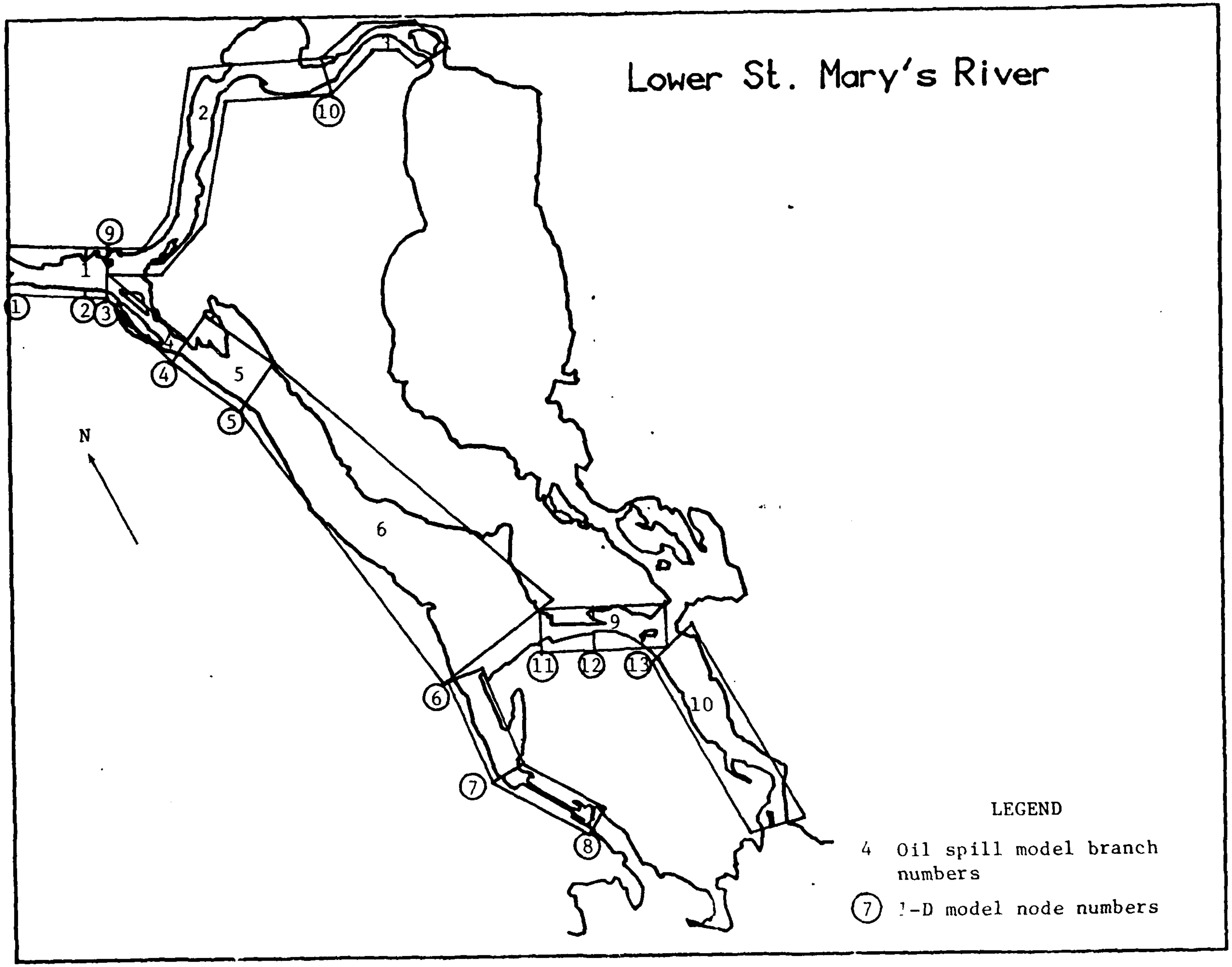


Fig. 6 Branch Configuration for Lower St. Mary's River

The assignment of velocities through the entire river plan area requires the establishment of a grid system, as shown in Fig. 7. The depth-averaged velocites obtained from the streamtube analysis are interpolated to all the grid boxes. The river boundaries for this grid system is defined by boundary grid boxes.

For each grid in the x-direction, two corresponding y grids for boxes at the upper and lower river shorelines are used to define the location of the river boundaries. When there is an island, two more y grids define the upper and lower boundaries of the island (see section on input output for more details). The model can handle any number of islands in the river. However, if two or more islands intersected a vertical line along which the x coordinate is constant, only one can be handled by the above mentioned input data. Other islands can be artificially handled by defining zero velocities for the island area (see section on input for more details). The disadvantage in this case is that oil deposition on shorelines may not be handled accurately.

Defining islands in the grid system is independent of defining islands for branch configuration. This allows for greater flexibility in the simulation of oil slick transformation. For example, a small island that covers about 3 or 4 grid boxes, which may be too small to be defined in the branch configuration, can be easily defined in the grid box system. The user must make sure that if an island is defined in the branch configuration, the area is appropriately defined in the grid box system as well.

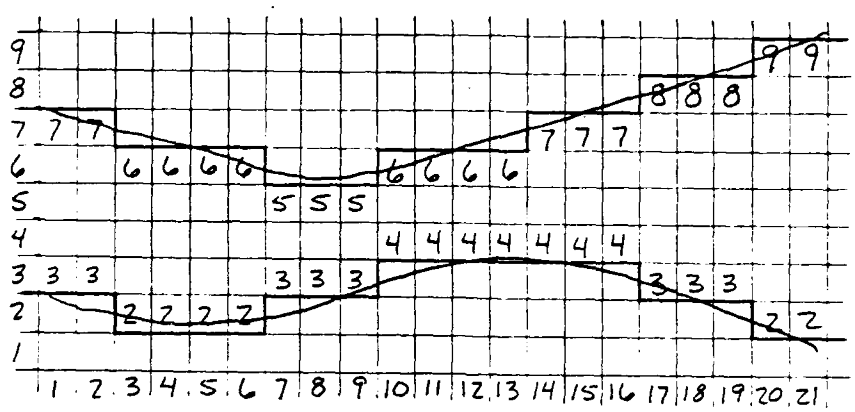


Fig. 7. Grid boxes and river boundary representation.

The velocity within each box is assumed to be constant. Only the boxes found within the defined river boundaries are assigned velocites. Generally, the spacing between cross sections in the stream tube computation area much larger than the grid spacing Δx. The following procedure is adopted for assigning velocities to grid boxes based on the velocities computed at cross sections. The description that follows use the term "velocity point" to represent a point at which a computed velocity is acting. First, all velocity points computed from the streamtube analysis will be assigned to the boxes within which they lie. If more than one velocity point fall into a box, the average of the velocities is assigned to that box. In the next step, extra velocity points will be obtained by interpolating between two successive cross sections. The number of interpolation points between two successive cross sections can be changed through user defined input data. These velocity points obtained through interpolation will now be assigned to grid boxes by using the same method as before. In the last stage if there are any boxes without an assigned velocity, an average velocity based on it's neighboring boxes will be assigned to them. Upon completion of these steps, the entire river is

scanned for boxes requiring assignment of a velocity, and a velocity is assigned based on the average velocity of the neighboring boxes.

Oil Slick Orientation

The oil slick can be expected to spread either axisymmetrically or in a one dimensional manner. Determination of which type of spreading prevails depends upon the shape of the slick (i.e. aspect ratio). Therefore, after the oil particles have been moved to a new position, the centroid of the oil slick, aspect ratio and orientation are calculated, but only from consideration of those particles which have **not** hit the boundary.

For the case where an island is encountered, particles may move along either side of the island. Since the model only handles one slick at a time, the particles moving along the top of the island are shifted to the bottom of the island by a distance equal to the island width. A single slick is now used for the determination of the slick orientation and aspect ratio. The shifted particles are moved back after the spreading phase has taken place in a separate subroutine.

Boundary Conditions

As discussed in Volume I, a parameter known as the "half life" is used to describe the ability of the shore to retain any oil. Currently, ten different half life values are built into the program. These values are:

CODE # -----------	1	2	3	4	5	6	7	8	9	10
Half Life (hrs) --	0.033	0.5	1	6	12	18	24	48	48	8760

For the ease of refering to a particular shoreline, the boundaries are designated according to Fig. 8.

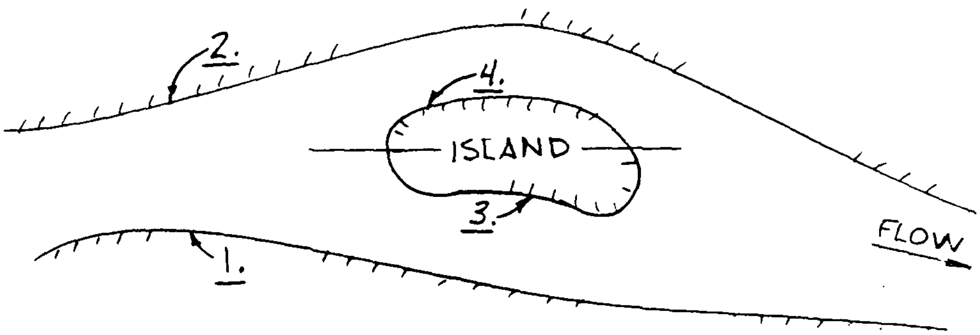


Fig. 8. Labeling of boundaries.

I.2 The Grid System

In the computer simulation, a two-dimensional reference is used to discretize the river. This reference is required to identify the locations of oil patches and their velocity. The finite-difference grids in this model must be squares of equal size. The location of each grid square is identified by their x and y indices. Detailed charts describing grid systems for all of the four connecting channels, with a grid size of 500 ft. x 500 ft., are given in Appendix II along with x,y indices. The user may refer to these charts to locate coordinates of oil spill site and locations of oil patches.

CHAPTER II

THE COMPUTER MODEL

The computer code has been tested on both FORTG1 and FORTRAN77 compilers. The listing given in this manual is the FORTRAN77 version. To run on FORTG1 the user needs to modify several statements in the main program.

All variables are dimensioned to have sufficient storage to run for any of the four rivers. However, the number of particles (NTOTAL) is a user defined input. If a number larger than 1000 is needed for NTOTAL, the dimensions of PARTCL, IMOVIN, YSHIFT, RADIUS and NTRACK must be appropriately increased.

II.1 Subroutine Cross Reference

A cross reference to the subroutines is provided to show the connection between the subroutines. Also, the associated input files are given where applicable.

Program(Main/Sub)	Calls	Input Files
ROSS	NDCONV, PRINT, VELDIS, PRELSE, ADVECT, ORIENT, SPRDAX, SPRD1Y, SPRD1X, EVAPOR, DISOLU, BOUNDR, PLOTNU	Geometry Spill data Shore conditions Ice Regions
ADVECT	GAUSS, RANDU*	none
BOUNDR	none	none
DISOLU	none	none
EVAPOR	none	none
NDCONV	none	flow data

ORIENT	none	none
PLOTNU	none	none
PRELSE	GAUSS, RANDU*	none
PRINT	none	none
SPRDAX	none	none
SPRD1X	none	none
SPRD1Y	none	none
VELDIS	none	none

*RANDU and GAUSS are standard subroutines available on IBM computers for generating uniform and gaussian random numbers respectively. On other systems they must be replaced by their equivalent.

COMMON BLOCKS

Common "VEL"

Variable Name	Algebraic Name	Type	Definition
CORDLB(I)		COMPLEX	lower bank coordinates of the I^{th} section
CORDV(I,J)		COMPLEX	coordinates at which VSTRM is acting
KINTM		INTEGER	number of interpolations between two velocity points of two consecutive cross sections, in the same streamtube.
LCSTSQ(I)		INTEGER	last cross section number of branch I
NFIRCO(I)		INTEGER	next cross section connecting to cross section in question. For a divided channel around island, this represents the first cross section connected to in the lower division from the main channel cross section
NSECO(I)		INTEGER	for a divided channel around an island, this represents the first cross section connected to in the upper division from the main channel cross section (if no island = 0, if lower division complete

		and returning back to upper division = 888, if both divisions are complete and resuming main channel = 999.)
NSLSCT(I)	INTEGER	number of sounding depths used to describe the channel geometry
NSTUBE(I)	INTEGER	total number of stream tubes at cross section I.
NUMCON(I)	INTEGER	Condition number of Section I. If all streamtubes continue to next cross section undivided = 11, if streamtubes divide into two channels from main channel = 12, if streamtubes from divided channel connect back to main channel = 21
Q(I)	REAL	discharge in the I^{th} branch
SCTANG(I)	REAL	angle I^{th} cross section makes with the positive x-direction. In the default version this angle is in radians. If it is needed to input in degrees, activate the comment statements in the DO loop in main program.
TICE(I,J)	REAL	equivalent ice thickness of J^{th} sounding depth in I^{th} cross section
VSTRM(I,J)	COMPLEX	stream velocity of the I^{th} cross section and J^{th} streamtube
WL(I)	REAL	water level at upstream end of branch I
YWID(I,J)	REAL	distance along the cross section from the reference bank to the J^{th} sounding depth in the I^{th} cross section
Z(I,J)	REAL	J^{th} sounding depth for the I^{th} cross section
ZD(I)	REAL	reference datum for section I from which the sounding depth is evaluated

Common "VA"

/ Variable Name /	Algebraic Name /	Type /	Definition /
VCAR(I)		COMPLEX	water velocity in grid box I of the cartesian system (Note: Two-D grid box system is converted to 1-D counting for storing)
VDRIFT		COMPLEX	drift velocity of oil
VWIND		COMPLEX	x and y components of wind velocity

Common "VASB"

/ Variable Name /	Algebraic Name /	Type /	Definition /
IGRILB(I)		INTEGER	y-direction grid box number of lower river boundary in column I (water side grid box)
IGRIUB(I)		INTEGER	y-direction grid box number of upper river boundary in column I (water side grid box)
IGRLB1(I)		INTEGER	y-direction grid box number of lower island boundary in column I (land side grid box)
IGRUB1(I)		INTEGER	y-direction grid box number of upper island boundary in column I (land side grid box)

Common "ASB"

/ Variable Name /	Algebraic Name /	Type /	Definition /
IHITB(I)		INTEGER	particle ID number or index of the I^{th} particle to hit the boundary
NHITB		INTEGER	number of particles which have hit the boundary.
NPTCL		INTEGER	number of particles in the system

PARTCL(I)	COMPLEX	cartesian coordinates of I^{th} oil particle
SPCEN	COMPLEX	cartesian coordinates of the center of the spill
TYPBND(I,J)	REAL	oil rejection rate from I^{th} shore in J^{th} grid box

Common "BLOCK7"

/ Variable Name	/ Algebraic Name	/ Type	/ Definition /
AK2I	k_{2i}	REAL	Fay's gravity-inertia phase spreading coefficient (axisymmetrical)
AK2T	k_{2t}	REAL	Fay's surface tension-viscous phase spreading coefficient (axisymmetrical)
AK2V	k_{2v}	REAL	Fay's gravity-viscous phase spreading coefficient (axisymmetrical)
ANIU	ν_w	REAL	kinematic viscosity of water (sq. ft. /sec.)
SIGMA	σ	REAL	surface tension of oil (lbs. /ft.)
SLICKR(I)		REAL	slick radius in I^{th} pie segment
SPGOIL		REAL	specific gravity of oil
VOLPAR		REAL	volume of one oil particle (cu. ft.)
VOLPIE(I)		REAL	volume of oil in the I^{th} pie segment (cu. ft.)

Common "BLOCK8"

/ Variable Name	/ Algebraic Name	/ Type	/ Definition /
AKC10	C_{10}	REAL	(Fay's or Waldman's) gravity - inertia spreading phase coefficient (one-dimensional)

AKC20	C_{20}	REAL	gravity - viscous phase spreading coefficient (one-dimensional)
AKC30	C_{30}	REAL	surface tension - viscous phase spreading coefficient (one-dimensional)

Common "SO"

Variable Name	Algebraic Name	Type	Definition
IMOVIN(I)		INTEGER	ID no. of the Ith moving particle
NMOVIN		INTEGER	total number of moving particles in system
SSHIFT		REAL	sum of all YSHIFT
YSHIFT(I)		REAL	distance I^{th} particle is shifted down from the upper island channel. (Applicable only when the spill is split by an island.)

Common "SE"

Variable Name	Algebraic Name	Type	Definition
FEVP1		REAL	fraction evaporated at the end of previous time step
FEVP2	F	REAL	fraction evaporated at the end of present time step.
CEVP	C	REAL	coefficient C (at $283^{o}K$) required for evaporation characteristics
TO	T_o	REAL	boiling point temperature of oil (^{o}K)

Common "ICE"

/ Variable Name /	Albegraic / Name	Type /	Definition /
NICERG		INTEGER	No. of ice regions
NICEX1(I)		INTEGER	x-grid box of the starting point of ice region I
NICEY1(I)		INTEGER	y-grid box of the starting point of ice region I
NICEX2(I)		INTEGER	x-grid box of the ending point of ice region I
NICEY2(I)		INTEGER	y-grid box of the ending point of ice region I
IPOS1(I)		INTEGER	to save space and time the program internally works with a 1-D array for storing most 2-D information, such as VELCAR( ). IPOS1(I) contains the 1-D array location corresponding to NICEX1(I), NICEY1(I).
IPOS2(I)		INTEGER	same as above (replace 1 by 2)
AMIUO	μ_o	REAL	viscosity of oil (gms/cm-sec)
ANICE	n_i	REAL	Manning's roughness coefficient for ice
SPAICE		REAL	Spill Area under ice cover (ft^2)

II.2 Main Program And Subroutines

ROSS

The main program ROSS is the main controlling code for the oil spill model. Through ROSS the fixed river geometry and oil spill parameter data are read into the program, variables are initialized and initial data manipulation and conversion is performed. Next, the appropriate subroutine calls are made to perform the required oil spill analysis and produce the final results as detailed in the Output section of each routine.

Input

The input data for ROSS, and the model in general, is found in four files. The file format is xxxx.yyy. The first four characters identify the river (e.g. STCL, DETR, STMU, STML). The last three characters are (i) GEO, (ii) FLW, (iii) BND, (iv) SPL and (v) ICE. Setup of input files is detailed in Chapter III.

Output

The output consists of printing of the following.

i.) Messages if errors are encountered during some checking procedures.

ii.) Headers, messages identifying spill type, spill location and spill duration.

iii.) Input parameters that describe spill material.

iv.) Stage and discharge conditions.

v.) Environmental conditions such as wind velocity and air temperature.

vi.) Locations of all particles (to a file) if this option is selected.

Common Blocks Referenced

VEL, VA, VASB, ASB, BLOCK7, BLOCK8, SE, V, SO, ICE

Subroutines Called

PRINT, VELDIS, ADVECT, ORIENT, PRELSE, SPRDAX,
SPRD1X, SPRD1Y, BOUNDR, PLOTNU, NDCONV, EVAPOR, DISOLU

Major Variables (not included in common blocks)

Variable Name	Algebraic Name	Type	Definition
API		REAL	API value of oil
DIFFUD		REAL	horizontal diffusion coefficient for the river (ft^2/s)
DX	Δx	REAL	size of grid box (ft)

FUELTP		CHARACTER	identification of fuel type
FULLTI		REAL	ice thickness of fully covered cross section, only **one** thickness assigned
HLIFE(I)	λ	REAL	halflife of oil spill retention on boundary, choice of one of ten possible values
HR		REAL	hydraulic radius of IY^{th} trapezoid in cross section
ICINFO		INTEGER	number of cross sections with ice covers
ICODE		INTEGER	integer identifying which of the ten halflife values to be assigned to a grid
IEVERY		INTEGER	frequency of obtaining output from **PLOTNU** and other subroutines (i.e. value of two (2)) gives output every other time step
INDPRN		INTEGER	two possible values: zero (0) results in no printout, one (1) results in printout
INDX1D		INTEGER	This variable can have values ranging from 0 to 5 (see also subroutine ORIENT). INDX1D = 0: axi-symmetrical spreading INDX1D = 1: 1-D spreading in y-dir; use SPRD1Y INDX1D = 2: 2-D spreading in x-dir; use SPRD1X INDX1D = 3,4, or 5 indicate that the slick is very short. If the slick is on free surface INDX1D will be set to INDX1D-3. If more than half the slick is under ice then INDX1D will be set to zero. After resetting INDX1D the procedures described above that correspond to 0, 1 or 2 will be followed.
IOPT1		INTEGER	two possible values: one (1) results in printout of fixed data like cross section geometry and shore conditions, zero (0) results in no printout
IOPT2		INTEGER	two possible values: one (1) results in output of computed velocities to be used for plotting, zero (0) results in no output
IOPT3		INTEGER	two possible values: one (1) results in output of particle locations to be used in plotting, zero (0) results in no output

IOPT4	INTEGER	two possible values: one (1) results in number plot of particle distribution (see **PLOTNU**), zero(0) results in no printout
IPARTX(I)	INTEGER	dummy variable used to transfer particle coordinates in integer format
IPARTY(I)	INTEGER	dummy variable used to transfer particle coordinates in integer format
IS	INTEGER	cross section number referencing ICINFO
ISPTYP	INTEGER	Spill Type; 0 - Instantaneous, 1 - Continuous. Computed by the model based on: if SPLTIM 0.5 * SPILDT, ISPTYP = 1, ELSE = 0.
IX	INTEGER	odd random number seed for random number generator
KINTM	INTEGER	no. of interpolations between two cross sections
LFROM	INTEGER	beginning limit (grid box number) for halflife designation to shore
LTO	INTEGER	ending limit (grid box number) for halflife designation to shore
NBRNCH	INTEGER	number of branches used in *oil spill* model
NBRP1	INTEGER	NBRNCH plus one
NGRIDX	INTEGER	total number of grid boxes in the x-direction
NTOTAL	INTEGER	total no. of paticles to be used
NTSTEP	INTEGER	total number of time steps for this run
OSTPS	INTEGER	No. of oilspill steps per UFDT
PERI	REAL	wetted perimeter of IY^{th} trapezoid in the cross section
REJRAT	REAL	oil rejection rate from boundary as calculated from HLIFE and SPILDT data
SOLBLT	REAL	solubility of oil (g/m^3)
SOLUNI	REAL	solubility of oil (lbs/ft^3)

SPAREA	REAL	spill area (ft^2)
SPCENO	COMPLEX	location of spill
SPILDT	REAL	magnitude of time step for spill simulation (seconds)
SPLRAT	REAL	rate of spilling (ft^3/s)
SPVOL	REAL	total volume of oil spill (U.S. gallons)
TENVF	REAL	air temperature (^oF)
THETA	REAL	wind direction, angle measured clockwise from north in degrees
THETAO(I)	REAL	the clockwise angle the y-axis makes with north in degrees for the four respective rivers (ex., THETAO(1) = 109.0)
TIMET	REAL	elapsed time of simulation (sec.)
TOTDIS	REAL	Total amount of dissolved oil (grams)
TOTIME	REAL	total time of oil spill simulation (sec.)
TTTT	REAL	elapsed time of simulation (hr.)
UFDT	REAL	1-D model time step (hrs.)
UFSTPS	INTEGER	No. of 1-D model steps
VMOL	REAL	molar volume of oil (m^3/mol.)
VMUNI	REAL	molar volume of oil (ft^3/mol.)
VWMAG	REAL	wind speed (ft/sec)
VWX	REAL	x-component of wind velocity (ft. /sec.)
VWY	REAL	y-component of wind velocity (ft. /sec.)
WNDSPD	REAL	wind speed (m/sec)
WORD	CHARACTER	cross section ice cover condition, "FULL" = fully covered, "PART" = partially covered, "OPEN" = open water

SUBROUTINE ADVECT

The subroutine **ADVECT** will utilize the wind and water velocities to calculate new positions of the oil spill particles. Two subroutine calls are made for each particle: one to **RANDU** to generate a uniformly distributed random number and one to **GAUSS** to generate a normally distributed (gaussian) random number. This information is used to calculate the turbulent fluctuation component of the water velocity.

Input

Data transferred into **ADVECT** from **ROSS** includes the spill simulation time step (SPILDT), grid box size (DX), particle ID no. from (N1) particle ID no. to (N2) to be considered for moving and random number generator seed (IX)

Information utilized by **ADVECT** includes:

1.) Locations of all particles.

2.) Wind and water velocities.

3.) Boundary information.

4.) Information on ice regions

Output

The new particle locations and the new number of particles which have hit the boundary (if any) are generated.

Procedure

1.) Check for particles which have hit the boundary (since only particles which have **not** hit the boundary will be advected.)

2.) Find the grid box where a particle is located.

3.) Determine whether the particle is under ice or not.

4.) Calculate the drift velocity for that grid box.

5.) Calculate the random velocity component and add to the drift velocity.

6.) Calculate the new position of the particle.

6.) Repeat 2-6 for each particle.

7.) Count the number of particles hitting the boundary after advection, and store their ID no's.

Common Blocks Referenced

VA, VASB, ASB

Subroutines Called

GAUSS, RANDU

Internal Variables

Variable Name	Algebraic Name	Type	Definition
ANG		REAL	angle of random component from the positive x-direction
DELEQ	δ_{eq}	REAL	equilibrium thickness required for ice covered case (ft)
DIFFUD	E_T	REAL	horizontal diffusion coefficeint (ft^2/s)
FRAMFA	k	REAL	friction amplification factor denoted by 'k' in text
IPOS		INTEGER	grid location in 1-D array VCAR
IX		INTEGER	random number generator seed
NPTCL		INTEGER	number of particles now in the system (Moving + Hit) NOTE: In the case of a continuous spill the no. of particles increase every time step for some period.
UFAIL	u_{f1}	REAL	failure velocity under rough ice cover (ft/sec)
UWATER		REAL	speed of water current (ft/s)
UTH	u_{th}	REAL	threshold current speed for slick movement (ft/sec)
VRAND		REAL	after return from **GAUSS,VRAND** = magnitude of random velocity component

VX	REAL	x-component of VRAND
VY	REAL	y-component of VRAND

SUBROUTINE BOUNDR

The subroutine **BOUNDR** handles the adsorption and rejection of oil at the river shorelines. This subroutine determines how many particles can remain on the appropriate land grid of a shoreline.

Input

Data transferred into **BOUNDR** from **ROSS** includes the grid box size (DX), the number of grid boxes in the x-direction (NGRIDX) and the printout indicator (INDPRN).

Information utilized by **BOUNDR** includes:

1.) Current particle locations including indices of particles which have hit the boundary.

2.) Boundary information.

Output

The output includes

1.) The locations of oil spill particles on the boundary.

2.) Recomputed oil volumes in boundary grid boxes.

3.) Relocation of rejected particles.

Procedure

1.) If a particle is below boundary one (1), move particle to appropriate boundary land grid.

2.) If a particle is above boundary two (2), move particle to appropriate boundary land grid.

3.) If neither 1.) or 2.) occurs, since these checks are performed only for hit particles, the particle is trapped between boundary three (3) and four (4). Therefore, assign particle to nearest island boundary.

4.) Repeat 1-3 for all particles which have hit the boundary.

5.) Check the boundary grid rejection rate and re-entrain the excess particles.

a.) First m particles of total m+n particles hitting the shore are retained and next n are pushed back out.

b.) If pushed out, particle is assigned to centroid of adjacent water grid.

6.) If particle is removed from shore, NHITB is reduced and remaining particle indices in array IMOVIN are shifted up to make up for the empty spot in the array.

7.) Repeat 5-6 for all x grid numbers checking each shore as required.

8.) Write output. (If NHITB $\neq$ 0)

Common Blocks Referenced

VASB, ASB

Subroutines Called

none

Internal Variables

Variable Name	Algebraic Name	Type	Definition
IALOWD		INTEGER	number of particles allowed in a grid box
IDUM1		INTEGER	temporary storage
IDUM2		INTEGER	temporary storage
J		INTEGER	temporary storage
K		INTEGER	temporary storage
K1		INTEGER	temporary storage
K2		INTEGER	temporary storage
L		INTEGER	temporary storage
M		INTEGER	temporary storage
NBNDR		INTEGER	boundary number(shore number)
NPTBND(K,I)		INTEGER	number of particles in boundary K of x-grid I.

X1	REAL	temporary storage
XCO	REAL	x-coordinate of the center of water grid adjoining first land boundary grid
XXX	REAL	used for checking location of particle in island
Y1	REAL	temporary storage
Y2	REAL	temporary storage
Y3	REAL	temporary storage
YCO	REAL	y-coordinate of the center of water grid adjoining first land boundary grid

SUBROUTINE DISOLU

The subroutine **DISOLU** computes the amount of oil dissolved in water. The solubility of oil is so low that it has very little affect on the trajectory (spreading), but it is important for environmental impact assessment. The working units in this subroutine are metric to make the cross reference with original theory easier.

Input

Data transferred to **DISOLU** from **ROSS** include spill area exposed to air (SPAREA,ft^2), spill area under ice (SPAICE, ft^2), solubility of fresh oil (SOLBLT,gram/m^3), time elapsed (TIMET,secs), Spill simulation time step (SPILDT,secs).

On Return

On return to main program it provides

1.) Amount dissolved during this time step (grams)

2.) Total amount dissolved (grams)

Grams are converted to lbs. in the main program.

Procedure

Theory by Cohen, Mackay and Shiu (1980) is used here. The subroutine

is self explanatory.

Common Blocks Referenced

ICE

Subroutines Called

None

Internal Variables

/ Variable Name	/ Algebraic Name	/ Type /	Definition /
ARBAR		REAL	Mean area of slick during the time step (m^2), ARBAR = (SPAR1 + SPAR2)/2
DELDIS		REAL	Amount of oil dissolved during the time step (grams)
DISOLK		REAL	Dissolution mass transfer coefficient, currently set at 1 cm/hr
SPAR2		REAL	total slick area at the end of present time step (in m^2) SPAR2 = (SPAREA + SPAICE)/10.76
SPAR1		REAL	Total slick area at the end of previous time step (m^2)
TOTDIS		REAL	Total amount of dissolved oil (grams)

SUBROUTINE EVAPOR

In this subroutine metric units are used. The reason for using units different from other subroutines is to make cross reference with theory (Mackay, et al. 1980) easier.

Input

Data transferred into EVAPOR from ROSS include API index of oil (API), Environmental temp (TENV, ^{o}K), Windspeed (WNDSPD, m/s), molar volume of oil (VMOL, m^3/mol), Initial volume of spill (VZERO, m^3), Spill area exposed to air (SPAREA, ft^2), Spill simulation time step (SPILDT) and step no. (JSTEP).

On Return

On return to main program it provides

1.) Fraction of oil evaoporated

Procedure

Theory by Mackay, Patterson and Nadeau is used here. The subroutine is self-explanatory.

Common Blocks Referenced

BLOCK7, SE

Subroutines Called

None

Internal Variables

/ Variable Name	/ Algebraic Name	/ Type /	Definition /
AKM	K_m	REAL	Mass Transfer coefficient (m/s)
C	C	REAL	Coefficient C at TENV
FEVP2		REAL	Fraction evaporated at present time step
JSTEP		INTEGER	Current Time Step
PO	P_o	REAL	Vapor Pressure at TENV (atm)

RGAS	R	REAL	Gas Constant e.g., 8.3147 Joules/deg mole
TENV	T_E	REAL	Air temperature (^oK)
WNDSPD	U_{wind}	REAL	Wind Speed (m/s)

SUBROUTINE NDCONV

This subroutine is necessary only if branch configuration in the oil spill model does not match exactly with 1-D flow model*. Currently the four rivers built into the model are: (i) St. Clair (IRCODE = 1); (ii) Detroit (IRCODE = 2); and (iii) lower St. Mary's (IRCODE = 3) and (iv) upper St. Mary's (IRCODE = 4). The branch configurations in oil spill model were slightly changed from those in the 1-D model to improve the computations of velocity distribution. In the case of upper St. Mary's River the branch configuration in 1-D flow model matches exactly with that in the oil spill model. Nevertheless, upper St. Mary's River is included in this subroutine for the convenience of the user. This subroutine also performs the function of reading in the water level and discharge data.

Input

Information utilized by NDCONV includes

1.) Water levels and discharge read as data

2.) River code (IRCODE)

On Return

On return, water levels and discharges at upstream and downstream ends of the branches in oil spill model are passed to the main program.

*If the branch configuration of oil spill model and unsteady flow model are indentical this program is not needed. See the section on DETR.FLW in Chapter III for more details.

Procedure

Conversion scheme is built into the subroutine through data in arrays RIV1, RIV2, RIV3 and RIV4. Depending on the value of river code the discharges and water levels will be assigned to match with the branch configuration of the river used.

Common Blocks References

VEL, VA

Subroutines Called

None

Internal Variables

Variable Name	Algebraic Name	Type	Definition
DWL		REAL	Temporary storage of water levels read in as data
DQ		REAL	Temporary storage of discharge read in as data
NPTS(I)		INTEGER	No. of data to read for river I
RIV1,RIV2,RIV3,RIV4		REAL	Transformation data for St. Clair, Detroit, upper St. Mary's and lower St. Mary's Rivers, respectively. Ex. RIV2(2) = 14 which means in Detroit River the 14th node from 1-D model corresponds to 2nd node of Oilspill Model.

SUBROUTINE ORIENT

The subroutine **ORIENT** calculates the oil slick orientation and aspect ratio. These values are then used to determine if axisymmetrical or one-dimensional spreading will be used.

Input

Data transferred into **ORIENT** from **ROSS** includes the grid box size (DX).

Information utilized by **ORIENT** includes:

1.) Current particle locations.

2.) Boundary information.

On Return

On return to main program it provides

1.) Oil slick orientation, aspect ratio and whether the slick is short, i.e., average radius DX/2

2.) Oil spill centroid.

3.) Locations of shifted particles from the upper side of island (if the slick is split by an island) and the distance that each particle has been shifted.

Procedure

1.) Find the particle ID numbers (indices) of those particles which have not hit the boundary. Assign these numbers to the array IMOVIN(I). [This information is used repeatedly, hence, this storing saves execution time.]

2.) Compute the spill centroid.

3.) If an island is encountered:

a.) Shift particles in upper channel to lower channel by a distance equal to the island width at x-coordinate of the particle while keeping track of the amount of shift and which particles were shifted.

b.) Recompute the spill centroid if particles were shifted.

4.) Calculate the angle of the slick between the principal axis of the slick and the river cartesian x-axis.

5.) Transform coordinates to have positions relative to the principal axis and compute aspect ratio.

6.) Using slick orientation (θ) and aspect ratio (a.r.) assign a value between 0 and 5 to INDX1D

INDX1D = 0 : a.r. ≤ 3

INDX1D = 1 : a.r. > 3, $0^\circ \leq \theta \leq 45^\circ$

INDX1D = 2 : a.r. > 3, $45^\circ < \theta \leq 90^\circ$

INDX1D = 3 : a.r. $\leq$ 3, ... short slick

INDX1D = 4 : a.r. > 3, $0^\circ \leq \theta \leq 45^\circ$, ... short slick

INDX1D = 5 : a.r. > 3, $45^\circ < \theta \leq 90^\circ$, ... short slick

Short slick is a slick having an average radius less than DX/2.

Common Blocks Referenced

SO, ASB, VASB

Subroutines Called

none

Internal Variables

/ Variable Name	/ Algebraic Name	/ Type /	Definition /
ASPECT		REAL	aspect ratio = SALONG/SNORMAL
BOT		REAL	temorary storage
COUNT		REAL	counter
CTHETA		REAL	cosine of the angle THETA
DEG		REAL	THETA in degrees
J		INTEGER	index for moving particles in IMOVIN(I)
L		INTEGER	temporary storage
M		INTEGER	temporary storage
SALONG		REAL	sum of particle distances from one of the transformed axis, as you move along the other axis
SNORML		REAL	sum of particle distances from the axis opposite to the one used to calculate SALONG
SPX		REAL	temporary storage
SPY		REAL	temporary storage
STHETA		REAL	sine of the angle THETA
SUMIX	I_y	REAL	sum of $(XX)^2$

SUMIXY	P_{xy}	REAL	sum of (XX) x (YY)
SUMIY	I_x	REAL	sum of $(YY)^2$
THETA		REAL	angle from cartesian x-axis to major spill axis
TOP		REAL	temporary storage
XX		REAL	x-distance from spill centroid to the particle
YY		REAL	y-distance from spill centroid to the particle

SUBROUTINE PLOTNU

The subroutine, **PLOTNU**, generates a numerical description of oil concentrations. This is accomplished by printing an area of size twenty columns of grids by twenty rows of grids. The printing area is centered over the oil spill centroid. If the grid box is included in or between the river boundaries, it will contain the number of oil particles currently contained there. If the grid box is not part of the river i.e. land, the grid box will be printed out containing "***". Since each column is formatted to be three characters wide, the maximum number of particles that will be printed out as a number for any given grid box is 999. If this number is exceeded, '***' will be printed for that grid box.

<u>Input</u>

Data transferred into **PLOTNU** from **ROSS** includes the grid box size (DX).

Information read into **ROSS** and utilized by **PLOTNU** includes:

1.) Current particle locations, spill centroid.

2.) River shore and island shore boundary information.

<u>Output</u>

Oil concentrations written as numbers of particles in a twenty by

twenty block of grid boxes centered over the current spill centroid. A sample output is given in Chapter III.

Procedure

1.) Assign all KOUNT a four digit number i.e. 1001 (note: format to write KOUNT is I3 so a four digit integer written in this format will appear as '***'.

2.) Find maximum and minimum x and y grid boxes (10 boxes away from spill centroid) and corresponding coordinates of the boxes centroid.

3.) Assign grid boxes (or KOUNT) which are contained within the river boundaries a concentration of zero (0) oil particles.

4.) Count how many oil particles in the grid boxes are contained within the grid box boundaries.

5.) Write the twenty by twenty block with the oil concentrations and the x and y ranges for the block.

Common Blocks Referenced

VASB, ASB, SO

Subroutines Called

none

Internal Variables

/ Variable Name	/ Algebraic Name	/ Type /	Definition /
IMAX		INTEGER	IMIN + 19
IMIN		INTEGER	x-grid box number 9 grids back from spill centroid.
JMAX		INTEGER	JMIN + 19
JMIN		INTEGER	y-grid box number 9 grids down from spill centroid.
KOUNT(1,J)		INTEGER	stores the number of particles in grid box
L,M		INTEGER	temporary storage
M1		INTEGER	lower grid box number describing river boundary

M2	INTEGER	upper grid box number describing river boundary
SPCEN1	COMPLEX	centroid of all moving particles computed based on their actual locations
XMIN	REAL	x-coordinate for centroid of leftmost grid boxes
XMAX	REAL	x-coordinate for centroid of rightmost grid boxes
YMIN	REAL	y-coordinate for centroid of upper grid boxes
YMAX	REAL	y-coordinate for centroid of lower grid boxes
XMIN1	REAL	minimum x-grid box number in twenty by twenty block centered over the spill
YMIN1	REAL	minimum y-grid box number in twenty by twenty block centered over the spill

SUBROUTINE PRELSE

The subroutine **PRELSE** is used to release particles in continuous spills at equal time intervals during the leak. For example, when a 30-minute spill is represented by a total of 600 particles, this subroutine will release a particle every 3 secs. Note that if **SPILDT** is 15 mins., it takes two time steps to release all the particles, with 300 particles released in each time step. Particles 1 to 300 have no effect from this subroutine during the second time step.

Input

Data transferred to **PRELSE** from **ROSS** includes grid box size (DX), spill simulation time step (SPILDT) random number generator seed (IX), particle ID no. from (N1), particle ID no. to (N2) to be considered for moving and initial spill site (SPCENO)

Information utilized by **PRELSE** includes

1.) Wind and water velocities

2.) Boundary information

3.) Information on ice regions

On Return

On return to the main program this subroutine provides the locations of released particles at the end of the time step and the number of particles that have hit the boundary (if any).

Procedure

1.) Find the grid box where the spill site is located.

2.) Determine whether that location is under ice or not.

3.) Calculate the drift velocity for that box.

4.) Calculate the random velocity component and add that to the drift velocity.

5.) Release a particle and move to it's location at the end of the time step.

6.) Advance time by (SPILDT)/(N2-N1+1).

7.) Repeat steps 4 to 6 for all particles from N1 to N2 (both inclusive).

8.) Count the number of particles hitting the boundary at the end of the time step, and store their ID no's.

Internal Variables

/ Variable Name	/ Algebraic Name	/ Type	/ Definition /
ANG		REAL	angle of random component from the positive x-direction
DELEQ	δ_{eq}	REAL	equilibrium thickness required for ice covered case (ft)
DIFFUD	E_T	REAL	horizontal diffusion coefficient for river (ft^2/s)
DTPTCL		REAL	time the particle has to travel from release time to the end of the current oil spill model time step

DTSMAL	δt	REAL	the smaller time step for advection computed based on stability criteria
FDELTA	F_δ	REAL	eq. (26), Vol. I
FRAMFA	k	REAL	friction amplification factor denoted by 'k' in text
IPASS		INTEGER	this is a counter to keep track of the loops within SPILDT
IX		INTEGER	random number generator seed
ROUGH		REAL	roughness height of ice (ft)
UFAIL	U_{f1}	REAL	failure velocity under rough ice cover (ft/sec)
UTH	u_{th}	REAL	threshold current speed for slick movement (ft/sec)
UWATER		REAL	water current speed (ft/s)
VRAND		REAL	after return from GAUSS,VRAND = magnitude of random velocity component
VX		REAL	x-component of VRAND
VY		REAL	y-component of VRAND

SUBROUTINE PRINT

The subroutine, **PRINT**, writes all fixed data describing the river configuration. This output could be directed to a terminal, printer or file. In most systems defining a value 6 will direct the output to a terminal or printer. Other values of IUT should be defined as a file. The fixed data is primarily the input for the computer model found in file XXXX.GEO and XXXX.BND as described in sections on **ROSS** and Chapter III. The choice of whether the output from **PRINT** will be written to a file or not depends upon the variable IOPT1. If IOPT1 equals one (1), information is written to a file, terminal or printer if IOPT1 equals zero (0) and no output is generated from **PRINT**.

Input

Data transferred into **PRINT** from **ROSS** includes the grid box size (DX), the unit number for printing the output (IUT), the number of branches (NBRNCH) and the number of grid boxes in the x-direction (NGRIDX).

Information read into **ROSS** and utilized by **PRINT** includes:

1.) River shore and island shore boundary information.

2.) Cross section locations, geometry and connection information.

3.) Boundary types and rejection rates.

Output

A heading and the fixed river configuration (geometry) is written to file OILPRT.OUT (if IUT = 2) and to the console if IUT = 6. Also included in the output is shore (boundary) type information. Value of IUT can be changed by changing the corresponding value in CALL PRINT statement.

Procedure

1.) Write the heading with date and time of program execution.

2.) Write number of branches, grid boxes in x-direction, grid box size and number of interpolations between sections.

3.) Write the information on sections for each branch.

4.) Write

   a.) cross section reference coordinates, orientation and width

   b.) number of streamtubes at that section

   c.) connecting conditions to the next streamtube

5.) Write the geometry of the cross sections i.e. The distance from the reference coordinates and corresponding sounding depth.

6.) Write the grid configuration for schematized river i.e. for every grid box in the x-direction, there exists an upper and lower y-grid for the river boundary and an upper and lower y-grid for an island. If no island, y-grids for island equal zero (0).

7.) Write rejection rates for each grid box.

Common Blocks Referenced

VA, VASB, ASB, VEL

Subroutines Called

none

Internal Variables

/ Variable Name	/ Algebraic Name	/ Type /	Definition /
DATRUN		STRING	system generated date of execution
KNUM		INTEGER	number of river boundaries equal 2 without islands, equals 4 with islands
IN		INTEGER	temporary storage
IS2		INTEGER	temporary storage
IWIDTH		INTEGER	temporary storage of YWID(I,J)
IUT		INTEGER	defines the unit number to which the output will be printed.
TIMRUN		CHARACTER	system generated time of execution

SUBROUTINE SPRDAX

The subroutine **SPRDAX** handles the axisymmetrical spreading of moving oil particles. The slick area is divided into eight pie segments and the axisymmetrical spreading equations are applied to particles in each pie.

Input

Data transferred into **SPRDAX** from **ROSS** includes the grid box size (DX), oil spill simulation time step (SPILDT), elapsed time of simulation (TIMET), the printout indicator (INDPRN), duration of the spill (SPLTIM) and volume rate of spill (SPLRAT).

Information utilized by **SPRDAX** includes:

1.) Current particle locations and spill centroid.

2.) Boundary information.

3.) Spreading law constants.

4.) Shifted particle information.

5.) Oil evaporation information.

6.) Information on ice regions.

Output

The following information is generated:

1.) New particle locations are calculated after spreading phase.

2.) Mean radius of slick in each pie is computed and printed

3.) Any shifted particles are placed back on the upper side of the island.

4.) Spill Area exposed to air, SPAREA, and spill area under ice SPAICE is computed.

Procedure

1.) Calculate the constant terms for spreading rate equations.

2.) Calculate the average radius of all moving particles.

3.) Locate numbers (indices) of particles which belong to a pie rejecting any outside of a calculated range.

4.) Calculate the mean pie radius.

5.) Determine whether the pie is under ice or not. If it is not under ice, skip Step. 6.

6.) Determine if the oil is still leaking. If leaking has stopped, no spreading, otherwise go to Step 9 and determine spreading under ice conditions.

7.) Calculate the transition times based upon the mean pie radius and eight times the oil volume in the pie segment.

8.) Determine the spreading phase of oil particles, and the mean pie radius.

9.) Determine the spreading rate.

10.) Calculate new particle locations.

11.) Calculate SPAREA and/or SPAICE

12.) Repeat 3-11 for each pie segment.

13.) Check to make sure no particles shifted from the top side of the island have spread through the island (if necessary).

14.) Shift particles back to upper side of island (if necessary).

15.) Check for particles hitting river or island shoreline.

Common Blocks Referenced

VASB, SO, BLOCK7, ASB, SE

Subroutines Called

none

Internal Variables

Variable Name	Algebraic Name	Type	Definition
AKINER		REAL	combined constants for gravity phase spreading rate equation
AKSURF		REAL	combined constants for surface tension phase spreading rate equation
AKVISC		REAL	combined constants for viscous phase spreading rate equation
ANG		REAL	angle a line that connects the particle location to spill center makes with positive x-direction
ANG1		REAL	upper angle limit of pie segment
ANG2		REAL	lower angle limit of pie segment
ATX1		REAL	temporary storage
ATX2		REAL	temporary storage
DELTA	$1 - \frac{\rho_o}{\rho_w}$	REAL	difference in specific gravity between oil and water
DRDT	dr/dt	REAL	axisymmetrical spreading rate at mean pie radius
DVDT	dv/dt	REAL	time rate of change of oil volume
G	g	REAL	gravitational acceleration

ICOND		INTEGER	ICOND = 0: oil in the pie has free surface conditions ICOND = 1: oil in the pie is under ice
IPIE		INTEGER	pie number
J		INTEGER	temporary storage of IMOVIN(I)
L		INTEGER	temporary storage
M		INTEGER	temporary storage
NPTPIE		INTEGER	number of particles in the pie, no more than 1000 particles allowed in a pie at one time
NTRACK(I)		INTEGER	keeps track of particle numbers in the pie
RADIUS(I)		REAL	distance to particles (radius) in the pie from spill center
RADOLD		REAL	radius before spreading (ft)
RADNEW		REAL	radius after spreading (ft)
RMEAN	$\bar{r}$	REAL	mean pie radius (ft)
ROWAT	ρ_w	REAL	density of water (slugs/ft^3)
SPRATE		REAL	spreading distance in SPILDT time (per unit radius)
SPX		REAL	temporary storage
SPY		REAL	temporary storage
TERMIN		REAL	time at which spreading terminates (sec)
TIMBAR		REAL	average of previous total simulation time plus current simulation time
TOTRAD		REAL	mean radius for all moving particles
TSURFT		REAL	transition time from viscous to surface tension phases (sec)
TVISC		REAL	transition time from irertia to viscous phases (sec)
VOLBAR		REAL	average of previous pie volume plus current pie volume
VOLNOW		REAL	current pie volume

X	REAL	temporary storage
Y	REAL	temporary storage

SUBROUTINE SPRD1X

The subroutine **SPRD1X** handles the one-dimensional spreading of moving oil particles. The slick is divided into strips and the one-dimensional spreading equations are applied to each individual strip.

Input

Data transferred into **SPRD1X** from **ROSS** includes the grid box size (DX), oil spill simulation time step (SPILDT), elapsed time of simulation (TIMET), the printout indicator (INDPRN), and spill area exposed to air (SPAREA).

Information utilized by **SPRD1X** includes:

1.) Current particle locations and spill centroid.

2.) Boundary information.

3.) Spreading law constants.

4.) Shifted particle information.

5.) Oil evaporation information.

6.) Information on Ice regions.

Output

The following information is generated:

1.) New particle locations are calculated after spreading phase.

2.) Mean position of the edge of the slick in each strip is computed and printed.

3.) Any shifted particles are placed back on the upper side of the island.

4.) Spill area exposed to air, SPAREA, and spill area under ice (SPAICE) is computed.

Procedure

1.) Calculate the constants needed for spreading rate equations.

2.) Locate the strip farthest downstream (LMAX) and the strip farthest upstream (LMIN).

3.) Calculate strip centroid with relation to y-coordinates of particles in strip.

4.) Determine particle ID numbers which belong to a strip, with a minimum of two particles required in a strip for spreading to take place.

5.) If more than half the particles in the strip are under ice then ICOND = 1; otherwise, ICOND = 0.

6.) If ICOND = 1, no spreading

7.) Calculate the mean spreading distance (width) of the slick in the strip (XLE) based upon the distance to particles from the strip centroid.

8.) Calculate the transition times based upon the oil volume in the strip.

9.) Determine in what phase oil particles at the mean width (XLE) are spreading and calculate the appropriate spreading rate.

10.) Calculate new particle locations.

11.) Repeat 7-10 for both sides of the slick centroid.

12.) Repeat 3-11 for each strip.

13.) Check to make sure no particles shifted from the top side of the island have spread through the island (if necessary).

14.) Shift particles back to upper side of island (if necessary).

15.) Check for particles hitting river or island shoreline.

Common Blocks Referenced

VASB, SO, BLOCK7, BLOCK8, ASB, SE

Subroutines Called

none

Internal Variables

Variable Name	Algebraic Name	Type	Definition
AKINER		REAL	combined constants for gravity phase spreading rate equation
AKSURF		REAL	combined constants for surface tension phase spreading rate equation
AKVISC		REAL	combined constants for viscous phase spreading rate equation
DRDT	dr/dt	REAL	one-dimensional spreading rate at mean strip width
DVDT	dv/dt	REAL	time rate of change of oil volume
ICOND		INTEGER	if ICOND = 0, the strip is treated as free surface. If ICOND = 1, the strip is treated as under ice
ISTRIP		INTEGER	strip number
LMAX		INTEGER	number of the strip farthest downstream
LMIN		INTEGER	number of the strip farthest upstream
NPT(I)		INTEGER	if I = 1, NPT is the number of particles on (+) side of YBAR; if I = 2, NPT is the number of particles on the (-) side of YBAR
NPTSTR		INTEGER	number of particles in a strip
NTRACK(I)		INTEGER	keeps track of particle numbers in the strip
RADIUS(I)		REAL	distances of particles from YBAR
SPRATE(I)		REAL	spreading distance in SPILDT time per unit distance to mean edge. If I = 1 it is for (+) side. If I = 2 it is for (-) side.
TERMIN		REAL	time at which spreading terminates (sec)
TIMBAR		REAL	average of previous total simulation time plus current simulation time (sec)
TSURFT		REAL	transition time from viscous to surface tension phases (sec)

TVISC	REAL	transition time from inertia to viscous phases (sec)
UTHICK	REAL	thickness at the temination of spreading (ft)
VOLBAR	REAL	average of previous strip volume plus current strip volume
VOLNOW	REAL	current strip volume
X	REAL	temporary storage
XLE(I)	INTEGER	mean strip width on (+) side of YBAR if I = 1 or on (-) side if I = 2
YBAR	REAL	centroid of strip width respect to the y-direction

SUBROUTINE SPRD1Y

This subroutine is very similar to **SPRD1X.** In this subroutine the strips are for a y-grid box, running in the direction of x-axis. The explanation is the same as **SPRD1X.**

SUBROUTINE VELDIS

The subroutine, **VELDIS,** calculates the two-dimensional depth-averaged velocity distribution in the river. The water velocity is determined for each grid box representing the river.

Input

Data transferred into **VELDIS** from **ROSS** includes the printing option (IOPT2), number of branches (NBRNCH), number of grid boxes in the x-direction (NGRIDX), and the grid box size (DX).

Information read into **ROSS** and utilized by **VELDIS** includes:

1.) branch connections, discharges and stages

2.) cross section locations, geometry, connecting information and ice thickness (if applicable)

3.) river shore and island shore boundary grid boxes

4.) information on the grid boxes to have zero velocities

Output

X and y components of velocity and coordinates where they act are computed and written (optional) to datafiles as follows:

a.) for each streamtube across all cross sections.

b.) for each grid box in the cartesian system representing the river.

Procedure

1.) For each branch:

a.) Go from cross section to cross section and calculate the streamtube velocities across the river width.

b.) Store the magnitude of the streamtube's velocity as complex variable VSTRM's x-component.

c.) Calculate the coordinates of the point at which the streamtube velocity acts.

2.) Calculate the direction in which the streamtube velocities act.

a.) Use the coordinates of velocities in the same streamtube for two consecutive cross sections and calculate the distance between them in x and y direction

b.) Use these distances to resolve the magnitude of streamtube velocity into x and y components by using similar triangles

3.) Print x and y velocity components and corresponding coordinates to a file if variable IPROPT = 1. (transferred from main program as IOPT2)

4.) The streamtube velocities computed at the cross sections are used to assign velocities to each grid box representing the river.

a.) Assign streamtube velocity to a grid box (across the cross section) if its coordinates are within a box (using averaging if two velocities fall into the same box.)

b.) Between two cross sections along the same streamtube, interpolate between the streamtube velocities at each section a set number (KINTM) of times using weighted averages and obtain KINTM interpolated velocities and corresponding coordinates.

c.) Using these interpolated velocities, assign velocities to the grid boxes between the cross sections using the same technique as described in a.)

d.) Search the river for grid boxes containing no velocities and assign velocities to them by taking an average of the velocities of the surrounding boxes.

5.) For the specified NZRVB boxes set velocities to zero

6.) If IPROPT =1, the x and y coordinates and the velocity assigned to these coordinates are printed to a file.

Common Blocks Referenced

VEL, VASB, VA, V

Subroutines Called

none

Internal Variables

Variable Name	Algebraic Name	Type	Definition
ANGL		REAL	temporary storage of SCTANG(I)
ARIY	$\sum^{P} A_p R_p^{2/3}$	REAL	area x (hydraulic radius)$^{2/3}$
ATUBE		REAL	cumulative area up to and including tube n
ATUBE1		REAL	cumulative area up to tube n
COUNT		INTEGER	number of boxes with assigned velocities surrounding the current box
DYRS		REAL	distance between sounding depth points defining the cross section
IBCON		INTEGER	next connecting branch
ICEIND		INTEGER	ice indicator; ice covered ICEIND = 1 for open water, ICEIND = 0
IPOS		INTEGER	keeps track of position in one-dimensional array for storing grid box velocities
IROW		INTEGER	used when checking to see if the island is inside the boundary or not

IS		INTEGER	cross section number
ISCON		INTEGER	temporary storage of cross section number of next connecting cross section
ITB		INTEGER	streamtube number
ITBCON		INTEGER	next connecting streamtube (island case)
IY1		INTEGER	lower y-direction grid box for a particular x-direction grid box
IY2		INTEGER	upper y-direction grid box for a particular x-direction grid box
J1		INTEGER	temporary storage of IY1
J2		INTEGER	temporary storage of IY2
L		INTEGER	x-direction grid box number
LASTSC		INTEGER	last cross section in branch
M		INTEGER	y-direction grid box number
MM		INTEGER	temporary storage
NFIRST		INTEGER	temporary storage of NFIRCO
NIY		INTEGER	number of sounding depths describing cross section geometry
NSTB		INTEGER	temporary storage of NSTUB
NSTUB1		INTEGER	NSTB - 1
PSARIY		REAL	partial sum of area x (hydraulic radius)$^{2/3}$
QIY	Q_p	REAL	cumulative discharge up to n^{th} sounding depth
QIY1		REAL	cumulative discharge up to $n-1^{th}$ sounding depth
QSET		REAL	set discharge for streamtube
QSTUBE	Q_s	REAL	computed discharge for any one streamtube
RAD		REAL	temporary storage
SAIY	$\sum^{P} A_p$	REAL	partial sum of areas

SARIY	$\sum^{N} A_p R_p^{2/3}$	REAL	total sum of area x (hydraulic radius)$^{2/3}$
SCTLEN		REAL	length from the beginning of a branch to a particular cross section in the branch. (For interpolation purpose only. Used only with TBRLEN.)
SPERI		REAL	sum of wetted perimeters
SXAREA	$\sum^{N} A_p$	REAL	total cross sectional area
TBRLEN		REAL	total branch length (For interpolation purpose only. Used only with SCTLEN.)
TISUM		REAL	sum of ice thicknesses at two successive points (sounding depth locations) across the cross section
VMAG	V_p	REAL	magnitude of velocity in streamtube
VVX		REAL	temporary storage
VVY		REAL	temporary storage
WLSCT		REAL	water level (stage) at cross section
X		REAL	x-coordinate of center of the box
Y		REAL	y-coordinate of center of the box
YSTB		REAL	distance from cross section reference point to center of stream tube
YSTB1		REAL	distance from cross section reference point to far side of streamtube
YSTB2		REAL	distance from cross section reference point to near side of streamtube

CHAPTER III

INPUT DATA FILES

There are two categories of input data that are required to run the model. The first category deals with what can be considered as fixed data. This is the information required to describe the river's shoreline and cross sectional geometry. Normally, there is no need to adjust this data once it has proven to give satisfactory results. The second category includes various parameters, constants, etc... which are to be adjusted depending upon the river discharge, spill location, spill type, spill volume, etc.

To completely understand the set up of the data files, it is helpful to go through the step by step procedure in the Section III.1. Sample input data files are given in Section III.3. This is especially important if the model is to be run for a river for which the data file has not been set up before. If the user is only interested in adjusting parameters, changing the spill location, or establishing new stage discharge conditions on a previously modeled river, section III.2 will be helpful in establishing the guidelines to follow. Note that formatting procedures for data input are not covered in this section, III.2.

III.1 Data File Creation

As a demonstration of the model's capabilities, the Detroit River will be used as an example in this Manual. Where the data files for Detroit River is not sufficient to describe the detail, other river data will be presented. This river was selected since it shows the complexity of river which the model can handle. All discussions on input and output data will refer to the Detroit River (Fig. 3.) from here on.

Five data files exist for inputting information into the computer

model. The first file **DETR.GEO** supplies all the necessary data for describing the river geometry and boundaries. The second file **DETR.FLW** contains discharge, elevation data at the nodes of one-dimensional model. The third file **DETR.BND** contains the half life data for the banks of the river. The fourth file **DETR.SPL** contains all data pertinent to the governing equations for the spreading of oil on the river surface. The fifth file **DETR.ICE**, contains the data describing the ice parameters and ice regions.

The first step in preparing the data file is to draw up a sketch similar to Fig. 9a and 9b. The intent of this sketch is to completely describe the branch and cross section numbering system used in the data file. The procedure to obtain this sketch is as follows:

1.) Determine the number of branches. Each branch must contain at least two cross sections. End branches (e.g. 15) must contain at least three cross sections.

2.) Number the cross sections in consecutive order from the upstream to the downstream end of the river. Around islands, number the bottom side up to the last cross section prior to convergence into a single channel then go back to the top side and continue the numbering sequence.

3.) Determine the number of streamtubes in each branch. The total number of streamtubes always remains constant so on each side of an island, the streamtubes must be divided up according to the ratio of the flow split around the island.

Once the branches and cross sections for the river is established, scaled maps of the river and cross sections are used to:

1.) Establish the overall x-y cartesian coordinates to be superimposed over the river.

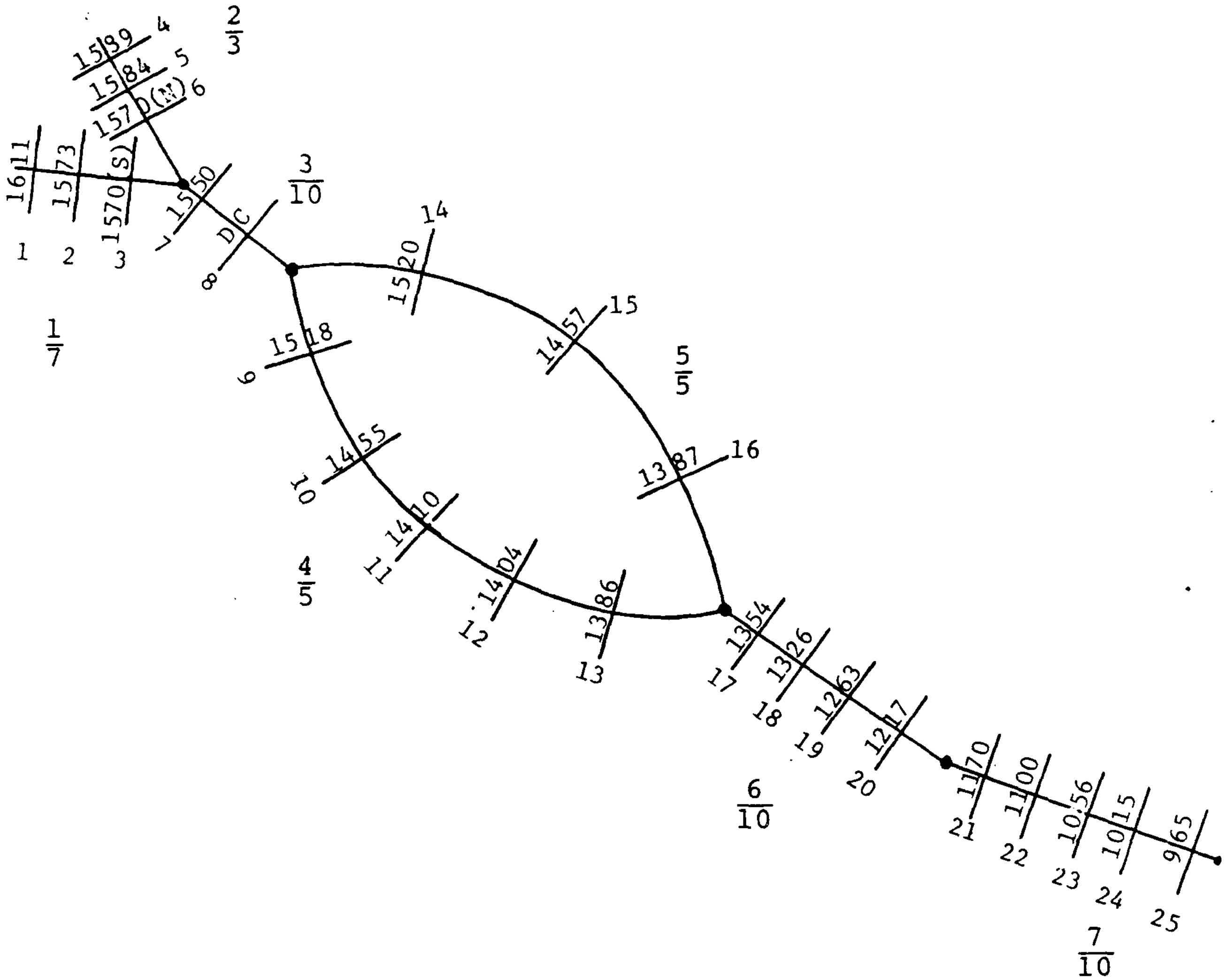


Fig. 9. Sketch of Detroit River with Numbering System

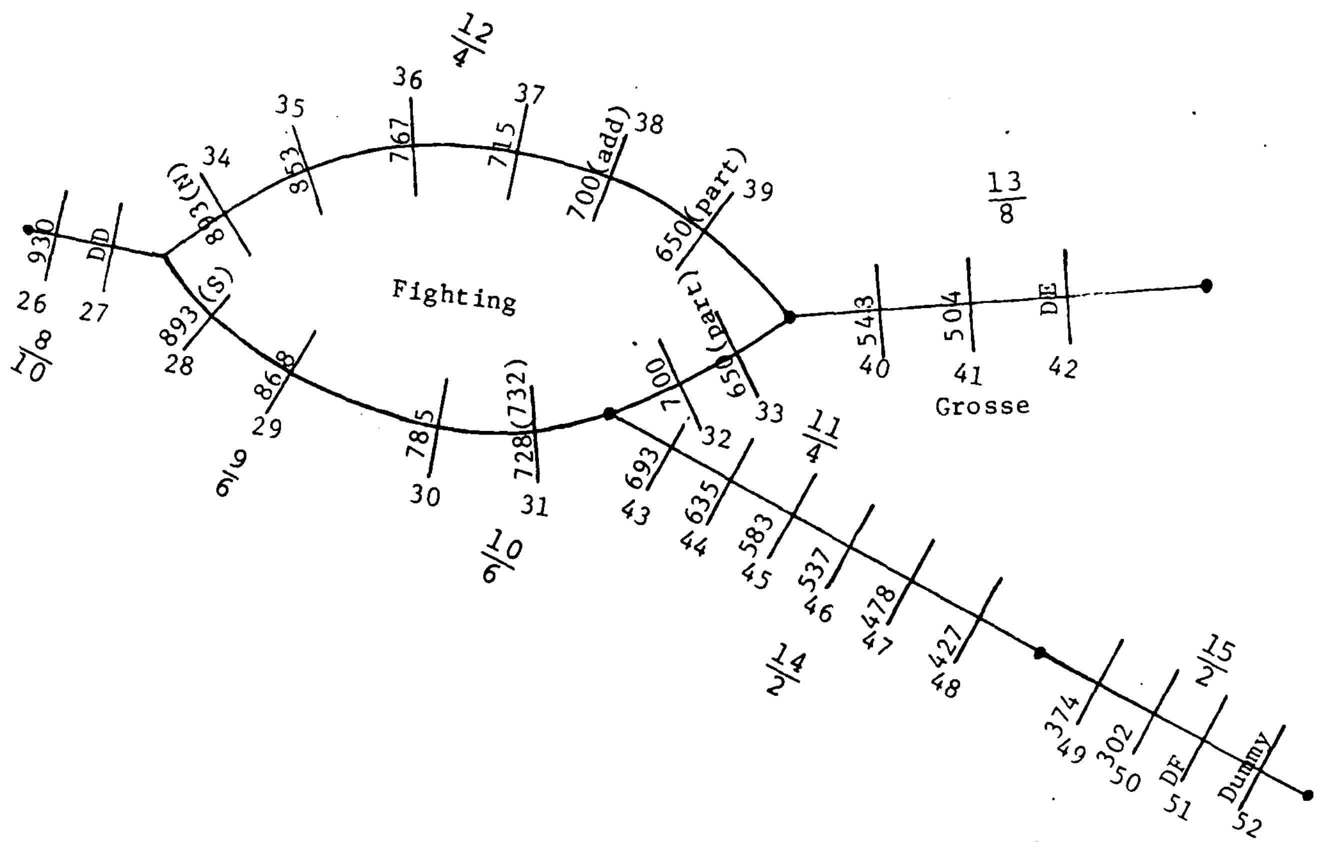


Fig. 9. Sketch of Detroit River with Numbering System

2.) Establish which side of the river the cross section locations (reference coordinates) will be referenced to.

3.) Digitize the entire river boundary and cross section locations simultaneously.

4.) Digitize the cross section geometries by measuring a distance from the reference coordinates to a corresponding sounding depth.

5.) Schematize the river into boundary boxes i.e. for every x-grid there exists corresponding upper and lower river (and island) shore boxes.

XXXX.GEO

The file DETR.GEO consists of five blocks of information with a varying or nonvarying number of cards in each. All blocks are listed below with the components and description. Most of the data read into the model is in list directed I/O (free format). If column numbers are shown, the data must be formatted accordingly, otherwise it is necessary to put only one space or comma between each number in a card. A sample DETR.GEO file is presented in Section III.3.

DETR.GEO; Block 1 -- branch and grid information

Card 1

example:

DETR Detroit River

Variable Name	Type and Length	Column Number	Definition
WORD	A4	1-4	keyword to define river. Example: STCL, DETR, STMU, STML

TEXT	19A4	5-80	Any text to identify the purpose for computer run.

<u>Card 2</u>

example:

15 250 500. 7

/ Variable Name	/ Type and Length	/ Column Number /	Definition /
NBRNCH	Integer	--	number of branches
NGRIDX	Integer	--	number of grid boxes
DX	Real	--	grid box size
KINTM	Integer	--	number of velocity interpolations between cross sections in a streamtube

<u>Card 3</u> (1 number for each branch)

example:

3 6 8 13 16 20 25 27 29 31 33 39 42 48 50

/ Variable Name	/ Type and Length	/ Column Number /	Definition /
LCSTSQ(I)	Integer	--	last cross section in each branch. Last branch - use second last cross section. There must be NBRNCH numbers (data). If line is not long enough, continue in the following line.

DETR.GEO; <u>Block 2</u> -- cross section location and connection information

<u>Card 1</u> (1 card for each cross section)

example:

1 (1895.4,31874.9) 0.80316770 9 11 2 0

/ Variable Name	/ Type and Length	/ Column Number /	Definition /
J	Integer	--	cross section number (for checking)
CORDLB(I)	Complex	--	complex variable giving x and y coordinate locating cross section on reference shore
SCTANG(I)	Real	--	angle (radians) cross section makes with positive x-axis
NSTUBE(I)	Integer	--	number of streamstubes at current cross section
NUMCON(I)	Integer	--	if all streamtubes continue to next cross section undivided = 11, if streamtubes divide into two channels from main channel = 12, if streamtubes from this channel and another channel connect to next section which is in main channel = 21
NFIRCO(I)	Integer	--	next cross section connecting to current cross section. For a divided channel around an island, this represents the first cross section connected to in the lower division from the main channel cross section
NSECO(I)	Integer	--	for a divided channel around an island, this represents the first section connected to in the upper division from the main channel cross section (if no island = 0. If this is first cross section in upper branch = 888. If this is last section in upper branch = 999.)

DETR.GEO; Block 3 -- cross section geometry

Card 1

example:

```
   3          9    574.91
```

/ Variable Name	/ Type and Length	/ Column Number	/ Definition /
J	Integer	--	cross section number (for checking)
NSLSCT(J)	Integer	--	number of sounding depths used to describe the cross section geometry
ZD(J)	Real	--	refernce datum for section J from which the sounding depth is evaluated

Card 2 (as many cards as required to input all sets of YWID,Z)

example:

```
  50.00  12.00   125.00  20.00  750.00  21.00  1250.00  22.00  1350.00  28.0
2425.00  28.00  2525.00   5.00 3250.00   4.00  3425.00   0.00
```

/ Variable Name	/ Type and Length	/ Column Number	/ Definition /
YWID(I,J)	Real	--	distance from the reference shore to the J^{th} sounding depth in the I^{th} cross section
Z(I,D)	Real	--	J^{th} sounding depth for the I^{th} section

NOTE: Block 3 must be repeated LCSTSQ(NBRNCH) times (i.e. = no. of cross sections defined).

DETR.GEO; Block 4 -- boundary grid boxes

Card 1 (1 for each grid in a x-direction)

example:

```
   14     54     74     64     65
```

/ Variable Name	/ Name ar / Length	/ Column Number	/ Definition /
J	Integer	--	x-grid box number

IGRILB(J)	Integer	--	y-direction grid box numer of lower river boundary for J^{th} x-grid (water side grid box)
IGRIUB(J)	Integer	--	y-direction grid box number of upper river boundary for J^{th} x-grid (water side grid box)
IGRLB1(J)	Integer	--	y-direction grid box number of lower island boundary for J^{th} x-grid (land side grid box)
IGRIB1(J)	Integer	--	y-direction grid box number of upper island boundary for J^{th} x-grid (land side grid box)

NOTE: Block 4 must be repeated NGRIDX (no. of grids in x-direction) times.

DETR.GEO; Block 5 -- Define any specific grid boxes to have zero velocity.

Card 1

example:

67

Variable Name	Type and Length	Column Number	Definition
NZRVB	Integer		No. of boxes to assign zero velocities

Card 2

example:

143 16 143 33 144 16

Variable Name	Type and Length	Column Number	Definition
IZRBX(I)	Integer		x-grid no. of Ith box to have zero velocity
IZRBY(I)	Integer		y-grid no. of Ith box to have zero velocity

There must be NZRVB pairs of IZRBX(I) and IZRBY(I). Data may be continued to as many lines as needed.

THIS IS THE END OF DATAFILE DETR.GEO.

XXXX.ICE

The DETR.ICE file contains information identifying ice regions which the user will have to adjust as ice conditions develop. An ice region is a range of grid boxes containing ice (Fig. 10). (Example, an ice region may be identified as extending from grid (15,7) to grid (18,12). The ice region then covers every grid from (15,7) to the upper shoreline of x column (15), all grids in X columns (16) and (17), and from the lower shoreline in x column (18) up to and including grid (18,12). In another example, an ice region may be identified as grid (21,7) to (21,9). Then, the ice region will only extend between y grids (7) and (9) in x grid column (21).) This information is used when determining if spreading and advection takes place under ice or on open water. A sample file listing is given in Section III.3.

DETR.ICE; Block 1

Card 1

Example:

0.035 12.5

Variable Name	Type and Length	Column Number	Definition
ANICE	REAL		Manning's n for ice roughness
AMIUO	REAL		viscosity of oil (gm cm/sec)(poise)

Card 2

Example:

1

Variable Name	Type and Length	Column Number	Definition
NICERG	INTEGER		total number of ice regions

Card 3 (1 card. If card is not long enough continue on next card)

Example:

15 7 18 9

Variable Name	Type and Length	Column Number	Definition
NICEX1(I)	INTEGER		x grid at the beginning of ice region
NICEY1(I)	INTEGER		y grid at the beginning of ice region
NICEX2(I)	INTEGER		x grid at the end of ice region
NICEY2(I)	INTEGER		y grid at the end of ice region

*NOTE: Cards 2 and 3 must be repeated for each time step in Unsteady Flow Model

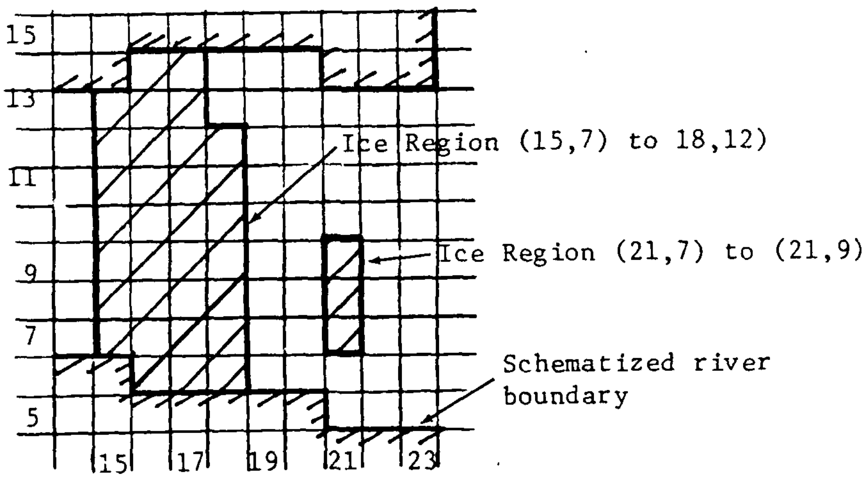


Fig. 10. Defining Ice Regions

XXXX.FLW

The DETR.FLW file contains the water level and discharge at each node in the river as defined by the one-dimensional flow model (Thomas, 1984). Also included are the ice conditions for each cross section in the river. This data is separate from the ice region data in DETR.ICE. The oil spill simulation model converts this information into boundary conditions for each river branch.

This file consists of three blocks of information. All blocks are listed below with descriptions and corresponding components. Blocks 2 and 3 must be repeated every time the velocities are updated in the model, i.e., every time step of the one dimensional flow model. Therefore, the data in this file needs to be adjusted on a more regular basis.

If the format and/or column numbers are shown, the data must be formatted accordingly, otherwise it is necessary to have only one space or comma between the data. A sample file listing of the DETR.FLW is given in Section III.3.

For St. Clair, Detroit and St. Mary's Rivers the difference in branch configuration between oil spill model and unsteady flow model is built onto the oil spill model through subroutine **NDCONV**. Hence, no. of branches for this datafile should be interpretted as the no. of branches in the unsteady flow model. If this model is used for other rivers the following steps are needed:

i) interpret no. of branches as the no. of branches in the oil spill model
ii) remove subroutine NDCONV
iii) activate the following three lines already in main program (presently given as comments)

```
DO 230 I = 1, NBRP1
  READ (7,*)WL(I),Q(I)
CONTINUE
```

DETR.FLW; Block 1 -- Time step in 1-D model.

Card 1 (one card)

example:

3.0

Variable Name	Type and Length	Column Number	Definition
UFDT	Real	--	Time in 1-D flow model in (Hrs.)

DETR.FLW; Block 2 -- discharge and storage

Card 1 (1 for each branch +1, the +1 comes due to downstream end)

example:

573.72   149190.

Variable Name	Type and Length	Column Number	Definition
WL(I)	Real	--	water level at upstream end of each branch (ft. above datum)
Q(I)	Real	--	discharge at upstream end of each branch (cfs)

DETR.FLW; Block 3 -- ice conditions (thickness information)

Card 1

example:

1

/ Variable Name	/ Type and Length	/ Column Number /	Definition /
ICINFO	Integer	--	number of cross sections with ice covered conditions. If no ice covered sections set ICINFO = 1 and then in Card 2, define the section to be open in the next card.

Card 2

example:

```
        2 OPEN
```

/ Variable Name	/ Type and Length	/ Column Number /	Definition /
IS	I4	1-4	number of cross section with ice covered conditions
WORD	A4	6-9	cross section ice cover condition, "FULL" = fully covered, "PART" = partially covered, "OPEN" = open water. If WORD = "FULL", then Card 3 has only one value and that is the ice thickness across the river for that x-section. If WORD = "PART", then Card 3 must have an ice thickness defined at each vertical line defining the x-section = NSLSCT(IS). If all numbers don't fit into one card as many cards as necessary may be used.

Card 3 (for fully covered cross section)
(NOTE: Card 3 must follow every Card 2 if WORD = "FULL" or WORD = "PART". If WORD = "OPEN", card 3 is not needed.)

example:

```
    0.6
```

/ Variable Name	/ Type and Length	/ Column Number /	Definition /
FULLTI	REAL	--	Ice thickness (ft) of fully covered cross section. Only one value is read as input and it will be assigned to the entire cross section.

Card 3 (for partially covered cross section)

example:

0.3 0.4 0.6 0.6 0.6 0.5 0.2

Variable Name	Type and Length	Column Number	Definition
TICE(I,J)	Real	--	Ice thickness (ft) of partially covered cross section. There must be one value for each sounding depth of the cross section with one additional for the extreme left of the cross section where depth sounding is not input through data because it is always assumed to be zero.

Block 2 followed by 3 must be repated for every 1-D model time step.

THIS IS THE END OF DATAFILE DETR.FLW.

XXXX.BND

The DETR.BND file consists of three blocks of information. All blocks are listed below with components and description. Most of the data read into the model are in free format. If the format and/or column numbers are shown, the data must be formatted accordingly, otherwise, it is necessary to have only one space or comma between the data. A sample file listing is given in Section III.3. All files with BND extension follow the same format.

DETR.BND; Block 1 -- half life data

Card 1 (1 card for each range of grid boxes)

example:

1 1 5 3

Variable Name	Type and Length	Column Number	Definition
K	Integer	--	Shore number. Lower y (river) = 1; upper y (river) = 2; lower y (island) = 3; upper y (island) = 4.
LFROM	Integer	--	beginning limit (grid box number) for half life designation to shore
LTO	Integer	--	ending limit (grid box number) for half life designation to shore
ICODE	Integer	--	integer identifying which of the ten half life values to be assigned to a grid

Card LAST (must be included).

example:

    0   0   0   0

XXXX.SPL

The DETR.SPL file consists of three blocks of information with a varying or nonvarying number of cards in each block. All blocks are listed below with the components and description. Most of the data read into the model are in free format. Guidelines for the length of the variables are given when necessary. If column numbers are shown, the data must be formatted accordingly, otherwise it is necessary to have only one space between each number in a card. A sample listing is given in Section III.3.

Block 1 -- oil spill and simulation parameters

Card 1 (Type of oil - Identification only)

Example:

    Fuel Oil No. 2

Variable Name	Type and Length	Column Number	Definition
FUELTP	CHARACTER	1-16	text for identifying the oil type

Card 2

example:

```
  24.0    1    0    1    0    0   1800.    -1.0
```

Variable Name	Type and Length	Column Number	Definition
TOTIME	Real	--	total time of oil spill simulation (hrs.). This value must equal or exceed the time step in unsteady flow model, i.e. in FLW file
IEVERY	Integer	--	frequency of obtaining output from **PLOTNU** and other subroutines. Ex., value of two (2) gives output every other time step
IOPT1	Integer	--	two possible values: one (1) results in output of fixed data such as cross section geometry and shore conditions, zero (0) results in no output
IOPT2	Integer	--	two possible values: one (1) results in output of computed velocities to a datafile to be used for plotting, zero (0) results in no output
IOPT3	Integer	--	two possible values: one (1) results in output of particle locations to a datafile to be used in plotting, zero (0) results in no output
IOPT4	Integer	--	two possible values: one (1) results in number plot of particle distribution (see **PLOTNU**), zero(0) results in no printout
SPLTIM	Real	--	Duration of oil spill in (secs), ex. oil might have been leaking for 15 mins. (i.e., SPLTIM = 900). This variable and SPLDT determines whether spill to be treated as continuous or instantaneous. SPLTIM = 0 is also allowed.
DIFFUR	Real	--	Horizontal diffusion coefficient (ft^2/s) for river. If the default formulation as described in Vol. I is desired set this value to -1.0

Card 3

example:

500 10000. 900. 0.84 1.411E-5 2.06E-3 1.14 .98 1.6 1.39 1.39 1.43

Variable Name	Type and Length	Column Number	Definition
NTOTAL	Integer	--	Total number of particles defined in the system (current maximum is 1000)
SPVOL	Real	--	total volume of oil spill (U.S. gal.)
SPILDT	Real	--	magnitude of time step for spill simulation (seconds)
SPGOIL	Real	--	specific gravity of oil
ANIU	Real	--	kinematic viscosity of water (sq. ft./sec.)
SIGMA	Real	--	surface tension of oil (lbs/ft)
AK2I	Real	--	Fay's gravity-inertia phase spreading coefficient (axisymmetrical)
AK2V	Real	--	Fay's gravity-viscous phase spreading coefficient (axisymmetrical)
AK2T	Real	--	Fay's surface tension-viscous phase spreading coefficient (axisymmetrical)
AKC10	Real	--	(Fay's or Waldman's) gravity - inertia spreading phase coefficient (one-dimensional)
AKC20	Real	--	gravity - viscous phase spreading coefficient (one-dimensional)
AKC30	Real	--	surface tension - viscous phase spreading coefficient (one-dimensional)

Card 4

example:

40000. 60000. .7063E-2 .1873E-2 7.88 465.0

Variable Name	Type and Length	Column Number	Definition
SPX	Real	--	x-coordinate of spill site
SPY	Real	--	y-coordinate of spill site
VMUNI	Real	--	molar volume of oil ft^3/mol.
SOLUNI	Real	--	solubility of fresh oil lbs/ft^3
CEVP	Real	--	coefficient C of evaporation characteristics of oil
TOEVP	Real	--	boiling point temperature of oil (^oK)

If you define a value of less than 1.0 for TOEVP the program defines the evaporation characteristics, using fitted curves. Therefore, the input values of CEVP and TOEVP have no influence on computations although they are read.

Block 2 -- wind and environmental temperature

Card 1 (1 card for each oil spill model time step)

example:

10.0 270.0 50.0

Variable Name	Type and Length	Column Number	Definition
VWMAG	Real	--	wind speed (ft/s)
THETA	Real	--	wind direction. Clockwise angle measured from north in degrees. ex., wind out of west = 270^o
TENVF	Real	--	air temperature in ^oF

II.2 Input Adjustments

For a river which already has the necessary input files, very little has to be modified to run the model under different conditions, i.e. new river discharge, different oil properties, new spill location etc. Conditions that are most likely to require modification is cited below with

some guidelines and suggested values for input. No attempt is made to explain the formatting of the data changes although references are made to the previous section "Create a Data File" concerning where the change is to be made.

Velocity

Assigning velocities to boxes has some dependency on the variable KINTM (DETRGEO, Block 1, Card 2). As KINTM increases, more velocity interpolations are made between cross sections. If the cross sections are spaced far apart, KINTM should be in the range of 5-8, otherwise KINTM can be 5 or less. The problem with using too small a value of KINTM is that more grid boxes will end up without assigned velocities after the interpolation stage. They will be assigned a velocity based on their neighboring boxes. This is less accurate than assigning a velocity based on interpolation between two cross sections. The target should be to get as many boxes as you can in the former stage. On the other hand a large value of KINTM will use more computing time.

Another factor which affects the assigning of velocities to grid boxes is the number of streamtubes selected. The choice of the number of streamtubes (NSTUBE, DETRGEO, Block 2, Card 1) depends upon the amount of computer time available and the accuracy desired. A river with highly irregular cross sections (versus nearly rectangular) or a desired high degree of accuracy in velocity computation will require the use of more streamtubes.

Stage Discharge

Stage and discharges (DETR.FLW, Block 2, Card 1) in river branches are unlikely to remain constant over time. Therefore this information will require updating as the need arises. The one-dimensional unsteady flow model can be used to obtain this information.

Oil Spill Parameters

The oil spill parameters (DETR.SPL, Block 1, Card 1,2,3,4) will require the most modification from spill to spill. Each spill will have its own simulation time, simulation time step, printing options, number of particles, volume, physical properties, and location, so it will be necessary to adjust this data each time the simulation is done.

Suggested guidelines for parameters:

NTOTAL is at least 500, maximum of 1000 is possible in the current version. The larger number will give a smoother result at the expense of longer execution time.

The time step suggested for SPILDT is 900 secs (15 mins.). However, this does not suit all situations. An example is a spill occurring near a narrow bend. In this case a smaller value for SPILDT is needed. A large value for SPILDT under these situations will lead to unrealistic results. On a different situation, where a spill occurs in a fairly straight and wide section of the river, one may use a larger value for SPILDT to speed up computation. This is a situation where the user is willing to sacrifice accuracy to obtain results quickly.

In the absence of data, the following values may be used.

SPGOIL = 0.7 to 0.98

ANIU = 1.411E-5, ft^2/sec.

SIGMA = 2.06E-3, lbs/ft.

AK2I = 1.14

AK2V = 0.98

AK2T = 1.6

AKC10 = 1.39

AKC20 = 1.39

ACK30 = 1.43

VMNUI = 0.7063E-2 ft^3/mol

SOLUNI = 0.1873E-2 lbs/ft^3

Wind Data

The computer model reads in wind data (DETR.SPL, Block 2, Card 1) at each time step of the simulation. This allows for forecasted wind speed and direction to be utilized by the model.

III.3 Sample Input Data Files

Sample data files included in this section are:

1. STCL.BND (Shoreline half life data file for St. Clair River)
2. LDETR.FLW (Flow data for Detroit River)
3. LDETR.GEO (Geometric data for Detroit River)
4. DETR.ICE (This file has the ice region information. The file is needed even when the model is run for open water conditions. Two samples are listed here; one corresponds to open water and the second one to when there is one ice region. Program allows you to have up to 20 regions.)
5. DETR.SPL (Spill information for Detroit River)

File STCL.BND

(This is the shoreline half-life data file for St. Clair River. STCL.BND is illustrated here instead of DETR.BND because DETR.BND is not very descriptive.)

```
1  1    3    7
1  4    36   3
1  37   37   3
1  38   62   3
1  63   66   3
1  67   85   3
1  86   87   7
1  88   97   3
1  98   98   7
1  99   188  3
1  189  190  7
1  191  192  3
1  193  193  7
1  194  219  3
1  220  220  7
1  221  223  7
1  224  237  3
1  238  239  7
1  240  259  3
1  260  261  7
1  262  264  3
1  265  267  7
1  268  291  3
3  86   89   7
3  90   92   10
3  93   93   3
3  94   98   10
3  99   105  7
4  86   86   7
4  87   89   3
4  90   90   7
4  91   91   3
4  92   99   3
4  100  105  7
3  226  231  7
4  226  227  10
4  228  231  7
4  5    151  3
2  152  156  10
2  157  160  3
2  161  166  10
2  167  240  3
2  241  242  10
2  243  247  3
2  248  250  7
2  251  271  3
2  272  274  10
2  275  291  3
0  0    0    0
```

File DETR.FLW

(This is the flow data file for medium flow conditions in Detroit River for open water conditions.)

```
  24.0
 573.72   149190.
 573.61   149180.
 573.61   184150.
 573.61   120810.
 573.46   120810.
 573.46   184140.
 573.14   184140.
 573.01   184140.
 573.01   153050.
 572.67   153050.
 572.67   115400.
 572.31   115400.
 572.31   146450.
 573.69    34970.
 573.61    34970.
 573.61    63340.
 573.46    63340.
 573.01    31070.
 572.31    31050.
 572.67    37640.
 571.97    37640.
 571.35    37630.
    1
    2 OPEN
 573.72   149190.
 573.61   149180.
 573.61   184150.
 573.61   120810.
 573.46   120810.
 573.46   184140.
 573.14   184140.
 573.01   184140.
 573.01   153050.
 572.67   153050.
 572.67   115400.
 572.31   115400.
 572.31   146450.
 573.69    34970.
 573.61    34970.
 573.61    63340.
 573.46    63340.
 573.01    31070.
 572.31    31050.
 572.67    37640.
 571.97    37640.
 571.35    37630.
    1
    2 OPEN
```

File DETR.GEO
(This is the geometric data file for the Detroit River)

```
DETR      Detroit River
15  250  500.   7
3  6  8  13  16  20  25  27  29  31  33  39  42  48  50
    1  (1895.4,31874.9)     0.80316770    9   11    2      0
    2  (3572.5,30227.7)     0.57252250    9   11    3      0
    3  (4329.2,29235.3)     0.85065230    9   21    7      0
    4  (6348.3,35251.1)     0.32890491    2   11    5    888
    5  (6573.7,34687.7)     0.27566774    2   11    6      0
    6  (6880.6,31783.6)     0.19895402    2   21    7    999
    7  (5218.9,28193.0)     0.60033042   11   11    8      0
    8  (6206.0,26230.4)     0.78546520   11   12    9     14
    9  (7367.3,25185.9)     0.38429453    4   11   10      0
   10  (11648.1,19916.6)    0.89099240    4   11   11      0
   11  (16161.9,17594.4)    1.06953870    4   11   12      0
   12  (16630.9,17489.5)    1.03244550    4   11   13      0
   13  (18885.3,16717.3)    1.29700210    4   21   17      0
   14  (10645.8,29130.4)    1.50359930    7   11   15    888
   15  (15909.5,24317.5)    0.89085370    7   11   16      0
   16  (19678.2,19845.3)    0.56874187    7   21   17    999
   17  (20184.2,16200.2)    0.65250602   11   11   18      0
   18  (23601.6,14140.2)    1.01761670   11   11   19      0
   19  (27986.0,10834.1)    0.84731720   11   11   20      0
   20  (32096.4,7511.7)     1.11644090   11   11   21      0
   21  (36555.3,5553.7)     1.31789400   11   11   22      0
   22  (43737.9,4722.7)     1.29165360   11   11   23      0
   23  (48111.9,4367.3)     1.73312770   11   11   24      0
   24  (52258.0,4974.1)     1.70696930   11   11   25      0
   25  (58891.5,6063.2)     1.74159230   11   11   26      0
   26  (60821.2,6100.4)     1.89850420   11   11   27      0
   27  (63193.4,5908.6)     1.85897050   11   12   28     34
   28  (64446.6,5679.9)     1.40498040    9   11   29      0
   29  (67199.6,4282.9)     1.55881210    9   11   30      0
   30  (75744.0,8630.3)     2.07408620    9   11   31      0
   31  (81028.9,9757.1)     2.01051160    9   12   43     32
   32  (83712.4,12984.0)    1.85562200    7   11   33    888
   33  (87888.6,14941.0)    1.97159570    7   21   40      0
   34  (63732.4,10102.1)    2.37748450    2   11   35    888
   35  (67304.8,12605.3)    1.95566550    2   11   36      0
   36  (75490.2,17058.0)    2.08554860    2   11   37      0
   37  (79227.9,19754.3)    1.69594960    2   21   38      0
   38  (82248.9,18595.8)    1.63638540    2   11   39      0
   39  (85910.8,19608.0)    1.96796920    2   21   40    999
   40  (95603.0,21152.6)    1.98250300    9   11   41      0
   41  (99230.8,23402.3)    2.09447510    9   11   42      0
   42  (101000.3,24459.3)   2.10296690    9   11   41      0
   43  (84015.6,10618.3)    1.97403760    2   11   44      0
   44  (89415.8,11292.4)    1.55666850    2   11   45      0
   45  (94336.8,12152.4)    2.10658900    2   11   46      0
   46  (98678.7,13600.2)    1.41654700    2   11   47      0
   47  (105015.3,14880.9)   1.97177820    2   11   48      0
   48  (109190.4,16653.8)   1.89456780    2   11   49      0
   49  (114885.3,17882.1)   2.00199130    2   11   50      0
   50  (121352.6,20917.5)   2.37169800    2   11   51      0
   51  (125494.9,23487.9)   2.67019810    2   11   50      0
     1       10  574.87
25.00   14.00  125.00   28.00  500.00   36.00 1000.00   37.00 1200.00   33.0
```

```
1750.00   32.00 2000.00   30.00 2250.00   10.00 3000.00    8.50 3425.00    0.0
      2      10  574.94
 155.00   25.00  900.00   27.50 1025.00   36.00 1275.00   40.00 1400.00   35.5
1700.00   43.00 2125.00   30.00 2375.00   10.00 3250.00    5.00 3375.00    0.0
      3       9  574.91
  50.00   12.00  125.00   20.00  750.00   21.00 1250.00   22.00 1350.00   28.0
2425.00   28.00 2525.00    5.00 3250.00    4.00 3425.00    0.00
      4       5  574.45
  75.00   12.00  175.00   22.00  250.00   27.00  825.00   27.00 1400.00    0.0
      5       5  574.50
  75.00   12.00  175.00   22.00  250.00   27.00  825.00   27.00 1250.00    0.0
      6      10  574.91
  90.80   11.00  141.60   26.30  197.80   28.80  373.00   29.50  627.40   30.8
 759.60   30.00 1123.80   24.10 1183.60   21.10 1233.10    9.50 1322.00    0.0
      7      15  574.72
 125.00   12.00  250.00   25.00  500.00   21.00  800.00   19.50 1000.00   27.5
1250.00   23.00 2050.00   22.50 2850.00   38.00 3500.00   38.00 4000.00   29.0
4250.00   43.00 4550.00   45.00 4925.00    6.50 5000.00    6.50 5100.00    0.0
      8      11  574.87
  75.00    4.00  275.00   18.00  575.00   24.00 1250.00   18.00 2075.00   12.6
2825.00   27.00 3825.00   29.00 4450.00   39.00 4600.00   18.00 4675.00    2.0
5075.00    0.00
      9       7  574.82
 101.10   25.00  266.10   24.80  506.50   29.10  756.20   33.00 1222.90   31.0
1360.00   25.50 1461.00    0.00
     10      11  574.54
  25.00   12.00   75.00   25.00  475.00   23.00  575.00   12.00  800.00    6.0
1250.00    2.00 1875.00    5.00 2050.00   22.00 2550.00   27.00 2750.00    5.0
3400.00    0.00
     11       7  574.27
  87.60   22.20  159.00   26.90 1106.90   20.90 1092.90   26.40 1997.50   23.4
2073.60   11.70 2110.70    0.00
     12      10  574.57
   1.90    9.50  135.00   25.40  550.10   30.50  695.30   24.30  882.50   29.0
1000.00   20.00 1373.70   22.30 1691.80   26.30 1843.60   21.10 1923.20    0.0
     13      11  574.16
  24.20   19.70  607.00   21.90 1004.50   28.50 1207.70   10.00 1388.10   32.7
1548.30   27.00 1684.20   19.90 1862.20   12.90 1891.40    7.30 2194.30    5.9
2203.80    0.00
     14      12  574.85
 125.00   25.50  250.00   35.00  350.00   33.00  680.00   47.50  825.00   42.0
1250.00   46.00 1525.00   35.00 1625.00   35.00 1825.00    9.00 2125.00    7.5
2533.00    9.00 2550.00    0.00
     15       8  574.18
 108.80   31.40  317.30   39.10  529.20   35.30  754.70   39.60 1003.90   36.6
1660.30   42.20 1831.30   24.10 1920.80    0.00
     16      12  574.47
  30.00    4.50   63.00   13.50  250.00   14.50  475.00   41.50  563.00   32.0
1125.00   32.50 1375.00   40.50 1625.00   42.50 1750.00   34.50 1825.00   38.0
1950.00   23.00 2000.00    0.00
     17      15  574.53
 125.00   12.00  375.00   18.00  425.00   25.00 1000.00   35.0[illegible] [illegible]50.00   37.0
1375.00   37.00 1500.00    9.00 1725.00    9.00 1850.00   25.00 2000.00   37.0
2775.00   29.00 2750.00   28.00  3125.00   32.00 3500.00   35.00 3750.00    0.00
     18       9  574.59
  10.00   27.00  550.00   28.00  625.00   37.00 1375.00   35.50 1750.00   36.0
```

```
2125.00   48.00 2310.00   43.00 2625.00   38.00 2910.00    0.00
     19       8  574.02
 117.60   26.20  368.50   21.70  634.20   31.80  811.10   50.80 1337.70   38.6
1657.20   48.00 2223.70   40.50 2372.80    0.00
     20      13  574.40
  25.00   23.00  125.00   27.50  200.00   23.80  475.00   34.00  600.00   29.5
 750.00   44.50 1000.00   49.80 1250.00   43.00 1600.00   47.40 1750.00   36.0
2000.00   45.00 2125.00   42.00 2250.00    0.00
     21       9  574.50
 125.00   18.00  175.00   29.00  375.00   41.00 1375.00   45.00 1500.00   47.0
1625.00   32.00 1800.00   18.00 1875.00    6.00 2050.00    0.00
     22       7  573.60
 197.00   37.50  451.10   46.20  717.30   46.90 1186.50   39.80 1593.60   42.8
1673.60   39.30 1903.10    0.00
     23       7  573.90
  25.00   29.00  125.00   38.00 1000.00   38.00 1500.00   31.00 2250.00   32.0
2500.00   27.00 2550.00    0.00
     24       8  574.19
  25.00    8.50  510.00   45.00 1120.00   37.50 2075.00   34.75 2375.00   37.5
2600.00   33.75 2710.00    8.00 2825.00    0.00
     25       9  574.14
  25.00   19.00  125.00   25.00  300.00   40.00 1000.00   39.00 1500.00   32.0
1750.00   36.00 2375.00   36.00 2625.00   24.00 2650.00    0.00
     26      11  574.12
 250.00   38.30  680.00   43.20 1125.00   36.00 1485.00   35.00 1750.00   40.8
2090.00   37.20 2500.00   37.50 2850.00   10.00 3320.00    5.20 3750.00    8.0
4250.00    0.00
     27       9  573.97
 175.00   34.00 1050.00   38.00 2175.00   28.00 2500.00   18.00 2675.00   25.0
3300.00   27.00 3425.00    4.00 4425.00    3.00 4675.00    0.00
     28      13  573.82
 185.00   28.50  375.00   40.00  500.00   41.00  625.00   40.20  850.00   35.5
1150.00   37.00 1500.00   35.80 1650.00   38.20 2000.00   29.50 2125.00    8.5
2310.00    4.80 2450.00    6.80 2510.00    0.00
     29      11  573.40
  27.10    3.20  230.90    6.00  450.10   20.00  835.60   30.00 1548.30   34.0
1842.40   45.50 2330.90   31.00 2586.60   34.40 2864.60    8.10 3354.00    6.5
3400.00    0.00
     30      14  573.39
 117.30   35.20  222.90   37.20  803.50   30.00  996.90   10.90 1503.20    6.0
1829.30   10.00 1975.90   31.00 2550.50   39.20 3196.90   35.00 3392.00   16.8
4179.00    4.80 4207.60    9.00 4276.50    8.20 4776.60    0.00
     31      19  573.51
  32.80   17.80  115.10   34.60  327.40   34.30  593.00   36.00  864.00    9.4
1148.70    6.40 1198.80    9.50 1254.40    0.00 1300.00    0.00 1337.40    7.9
2000.90   32.80 2389.90    7.20 2443.4     6.20 2983.60   13.70 3234.50   38.0
4076.00   38.00 4409.10    9.30 6203.8     6.00 6715.40    6.40 6760.70    0.0
     32      12  573.75
 300.00   31.00  500.00   31.00  780.00    4.50 1310.00    4.60 1790.00   35.0
2090.00   37.00 2900.00   35.00 3250.00   36.00 3680.00    7.20 5600.00    6.8
5650.00    9.00 5750.00    0.00
     33      10  572.40
 450.00    6.00  575.00   28.00 1250.00   36.00 1750.00   33.00 2700.00   28.0
2725.00   18.00 2825.00    5.00 4500.00    3.00 5000.00    2.00 5025.00    0.0
     34       8  573.73
 120.00    9.10  174.00   19.60  451.30   34.90  851.00   32.00  986.20   21.9
```

1040.00	13.40	1106.00	5.30	1175.30	0.00			
35	5	573.30						
24.40	14.40	573.00	33.40	722.30	31.00	896.10	16.00	975.80	0.0
36	8	573.40						
198.80	23.60	501.90	25.50	853.90	24.10	952.30	8.60	1887.40	7.0
1980.10	25.80	2265.30	21.40	2428.00	0.00			
37	9	573.68						
92.20	24.90	301.20	31.00	445.60	28.10	680.60	35.20	969.40	22.0
1051.70	8.10	1707.00	7.20	2242.90	4.30	2375.20	0.00	
38	11	573.76						
1500.00	0.01	1625.00	25.00	2000.00	25.00	2100.00	2.00	2300.00	0.0
3750.00	0.00	3825.00	5.00	3900.00	32.00	4225.00	32.00	4375.00	2.0
4675.00	0.00							
39	10	573.83						
25.00	2.00	625.00	1.00	750.00	27.00	1125.00	27.00	1175.00	32.0
1250.00	30.00	1750.00	32.00	1925.00	6.00	2125.00	2.00	2150.00	0.0
40	17	573.91						
50.00	3.00	475.00	3.00	1000.00	26.50	2000.00	27.20	2100.00	35.0
2800.00	35.00	3100.00	24.80	3300.00	32.00	3750.00	30.00	4250.00	33.0
4500.00	27.80	5100.00	28.00	5500.00	9.00	5900.00	7.50	6900.00	8.0
7000.00	12.00	7100.00	0.00					
41	15	572.64						
72.80	3.30	292.90	6.40	2564.00	10.70	2709.50	26.60	3745.50	24.4
3771.70	28.70	4424.00	36.20	4509.30	27.30	4996.00	25.00	5150.60	22.2
5327.90	27.80	5482.00	28.30	5796.10	11.00	6110.80	13.10	6345.80	0.0
42	12	572.70						
575.00	6.00	750.00	6.00	1000.00	5.00	2750.00	3.00	3175.00	6.0
3375.00	18.00	4300.00	18.00	4375.00	27.00	4950.00	27.00	5400.00	18.0
5900.00	6.00	6050.00	0.00					
43	8	573.76						
16.70	4.70	129.10	30.60	283.70	36.10	474.10	36.90	657.20	30.0
812.00	8.70	866.90	7.40	898.10	0.00			
44	10	573.23						
4.00	3.00	44.90	3.50	243.80	32.30	536.30	33.00	779.30	31.4
992.50	6.70	1152.00	4.30	1849.90	3.30	1940.20	1.90	2002.80	0.0
45	11	573.45						
55.40	19.00	277.50	18.00	433.60	23.00	460.00	15.10	501.60	32.6
772.50	34.60	822.90	21.00	848.20	25.30	978.40	10.50	1095.20	7.5
1246.00	0.00							
46	10	572.93						
15.70	2.20	70.40	2.50	239.90	21.90	439.70	9.50	590.60	9.7
876.40	27.30	1237.90	24.30	1376.90	7.70	1496.30	4.50	1523.20	0.0
47	10	573.12						
47.90	3.60	99.50	23.90	636.30	16.10	756.00	19.90	837.70	17.6
981.80	29.60	1143.90	26.50	1165.10	20.00	1379.10	14.70	1425.00	0.0
48	4	572.83						
31.30	25.20	209.10	29.00	990.20	27.10	1145.90	0.00	
49	10	572.82						
226.10	7.50	274.60	6.50	766.00	7.80	1245.50	21.80	1345.30	19.1
1805.80	15.70	1921.90	20.40	2395.50	15.50	2471.90	9.80	2638.50	0.0
50	18	572.68						
319.50	19.60	405.70	8.20	559.00	9.30	881.90	22.80	10[illegible].40	7.9
1508.20	5.40	1676.10	21.90	2128.40	15.00	2194.10	9.00	2468.20	9.7
2542.30	17.20	2646.40	13.60	2708.50	7.80	2770.80	7.10	3015.40	1.9
3208.80	5.90	3431.90	7.20	3674.50	0.00			
51	11	572.60						

File DETR.GEO (cont.)

```
 875.00     4.00 1000.00    18.00 2000.00    18.00 2175.00     5.00 2550.00     0.0
4125.00     0.00 4625.00     2.00 4875.00    14.00 5125.00    14.00 5500.00     4.0
6500.00     0.00
  1   65  135    0    0
  2   65  126    0    0
  3   65  117    0    0
  4   65  108    0    0
  5   64   99    0    0
  6   63   91    0    0
  7   62   90    0    0
  8   61   89    0    0
  9   60   88   70   71
 10   59   86   69   72
 11   57   85   69   72
 12   56   83   68   72
 13   55   80   66   71
 14   54   74   64   65
 15   52   72    0    0
 16   50   70    0    0
 17   48   64   54   56
 18   46   63   52   57
 19   45   62   50   57
 20   44   61   48   56
 21   43   61   47   56
 22   42   61   47   56
 23   42   61   47   56
 24   41   61   46   56
 25   40   61   46   56
 26   39   61   46   54
 27   38   60   46   53
 28   38   60   46   53
 29   38   58   45   51
 30   37   57   45   50
 31   37   55   44   50
 32   36   54   43   49
 33   36   53   42   48
 34   36   52   41   47
 35   36   52   40   46
 36   36   51   39   46
 37   35   50   39   44
 38   35   49   39   43
 39   34   48   39   41
 40   34   47   39   39
 41   33   46    0    0
 42   33   44    0    0
 43   32   42    0    0
 44   32   40    0    0
 45   31   39    0    0
 46   30   38    0    0
 47   30   37    0    0
 48   29   36    0    0
 49   28   35    0    0
 50   29   34    0    0
 51   27   33    0    0
 52   26   32    0    0
 53   25   31    0    0
```

File DETR.GEO (cont.)

54	25	30	0	0
55	24	29	0	0
56	23	28	0	0
57	22	27	0	0
58	21	26	0	0
59	21	26	0	0
60	20	25	0	0
61	19	24	0	0
62	18	23	0	0
63	17	22	0	0
64	16	21	0	0
65	16	20	0	0
66	15	20	0	0
67	15	19	0	0
68	15	18	0	0
69	14	18	0	0
70	14	17	0	0
71	13	17	0	0
72	13	16	0	0
73	12	16	0	0
74	12	15	0	0
75	12	15	0	0
76	12	15	0	0
77	11	15	0	0
78	11	14	0	0
79	11	14	0	0
80	11	14	0	0
81	11	14	0	0
82	11	14	0	0
83	11	14	0	0
84	11	14	0	0
85	11	14	0	0
86	11	14	0	0
87	11	14	0	0
88	10	13	0	0
89	10	13	0	0
90	10	13	0	0
91	10	13	0	0
92	10	13	0	0
93	10	13	0	0
94	10	13	0	0
95	10	14	0	0
96	10	14	0	0
97	10	14	0	0
98	10	15	0	0
99	10	15	0	0
100	10	15	0	0
101	11	15	0	0
102	11	15	0	0
103	11	15	0	0
104	11	16	0	0
105	11	16	0	0
106	11	16	0	0
107	11	16	0	0
108	12	16	0	0
109	12	16	0	0

110	12	17	0	0
111	12	17	0	0
112	13	17	0	0
113	13	17	0	0
114	13	18	0	0
115	13	18	0	0
116	13	18	0	0
117	13	18	0	0
118	13	19	0	0
119	13	19	0	0
120	13	20	0	0
121	13	21	0	0
122	13	21	0	0
123	13	21	0	0
124	13	21	0	0
125	13	22	0	0
126	13	23	0	0
127	13	24	0	0
128	13	25	19	21
129	13	26	19	22
130	12	27	18	23
131	12	27	18	24
132	11	28	18	25
133	11	28	17	25
134	10	28	17	25
135	10	30	18	25
136	9	31	18	26
137	8	31	19	26
138	8	32	20	26
139	8	32	20	26
140	8	32	21	27
141	8	33	22	27
142	8	34	22	28
143	9	35	23	29
144	9	36	24	29
145	10	36	24	30
146	11	37	25	31
147	12	37	26	31
148	13	38	27	32
149	15	38	28	32
150	18	39	29	33
151	18	40	30	34
152	18	41	30	35
153	18	42	31	36
154	19	42	31	36
155	19	43	32	37
156	19	43	33	38
157	20	43	33	39
158	20	44	34	39
159	20	44	34	40
160	20	45	35	40
161	20	45	36	40
162	20	46	23	23
163	21	47	23	24
164	21	46	23	24
165	21	45	23	24

166	21	46	23	25
167	22	46	24	25
168	22	46	24	26
169	22	45	24	27
170	23	44	24	27
171	23	47	25	28
172	23	47	25	28
173	24	48	25	29
174	24	48	25	29
175	24	48	26	29
176	24	48	27	30
177	24	48	27	30
178	24	52	27	30
179	24	52	27	31
180	23	52	27	32
181	23	55	27	33
182	23	55	27	35
183	23	55	26	36
184	23	55	26	37
185	23	55	26	38
186	23	55	26	39
187	23	55	26	40
188	24	56	27	41
189	25	56	27	41
190	25	57	28	42
191	26	57	29	42
192	26	57	29	43
193	27	58	30	43
194	27	58	30	44
195	27	59	30	44
196	28	59	30	45
197	28	60	30	45
198	28	60	31	46
199	28	61	31	47
200	28	62	31	47
201	28	62	31	48
202	27	63	31	48
203	28	64	31	49
204	28	50	32	50
205	29	50	32	50
206	29	50	32	50
207	30	51	32	51
208	30	51	33	51
209	30	51	33	51
210	31	51	34	51
211	31	52	34	52
212	31	52	35	52
213	32	55	35	55
214	32	53	36	53
215	33	53	36	53
216	33	53	36	53
217	34	54	36	54
218	34	54	37	54
219	34	54	37	54
220	35	54	37	54
221	35	55	38	55

```
222   35   55   38   55
223   36   56   39   56
224   36   56   40   56
225   36   57   40   57
226   36   57   41   57
227   35   58   41   58
228   36   58   42   58
229   35   58   42   58
230   37   58   43   58
231   37   59   43   59
232   38   60   44   60
233   38   60   45   60
234   38   60   45   60
235   38   60   45   60
236   38   60   45   60
237   38   60   47   60
238   39   60   48   60
239   39   59   49   59
240   40   53   51   53
241   41   53    0    0
242   42   52 .  0    0
243   43   51    0    0
244   44   51   50   51
245   44   52   50   52
246   45   53   50   53
247   46   55   50   55
248   47   56   52   56
249   47   57   54   57
250   48   58   55   58
67
138 9   138 10   138 29   138 30   139 30   139 31   140 30   140 31   141 32   142 32
143 16  143 33   144 16   144 17   144 18   144 34   145 16   145 17   145 18   145 35
146 18  146 19   146 36   147 19   148 19   148 20   149 19   162 36   162 39   162 42
163 36  163 42   163 43   164 36   164 37   164 41   164 42   164 43   164 44   165 36
165 37  165 38   165 41   165 42   165 43   165 44   166 37   166 38   166 42   166 43
166 44  167 38   167 39   167 43   167 44   168 39   168 43   169 40   170 40   171 44
171 45  172 44   173 44   173 45   174 44   174 45   175 45
```

File DETR.ICE

(This file can be used for any river if open water conditions exist.)

```
 0.035     0.34
0
```

File DETR.ICE

(This file shows the data arrangement when there is an ice region in the river.)

```
 0.035     0.34
1
9  19  122  21
```

File DETR.SPL
(This is the spill information file for the Detroit River)

```
No. 2 Fuel Oil
  6.0   1   0   0   1   0  10.    -1.0
500  10000.  900.  0.84  1.411E-5  2.06E-3 1.14  0.98  1.6  1.39  1.39  1.43
40000.    6000.   .7063E-02  .1873E-02   7.88   465.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
5.866   270.0   50.0
```

CHAPTER IV

MODEL OUTPUT

Output from the model can be directed to different devices, i.e. console, printer and files.

Included in this chapter are:

1. The output generated as a file by subroutine PRINT for St. Clair River.

2. The graphical output of velocity distribution corresponding to low flow ($Q \approx 130{,}000$ cfs) in St. Clair River as generated by program VELDIS.GRP. Input data for these plots were generated by ROSS.

3. A sample computer printout generated by an instantaneous spill of 5000 gallons of No. 2 fuel oil. Wind is 2 mph from West. Air temperature = $70^{o}F$.

4. Graphical output that corresponds to above mentioned simulation.

1

```
**************************************************************************
**************************************************************************
**                                                                      **
**               SIMULATION MODEL FOR OIL SPILLS IN RIVERS              **
**                                                                      **
**     DEVELOPED AT - CIVIL & ENVIR. ENG. DEPT., CLARKSON UNIVERSITY     **
**     SPONSERED BY - U.S. ARMY CORPS OF ENGINEERS, DETROIT DISTRICT    **
**                                                                      **
**       DATE AND TIME OF RUN : THU SEP 26, 1985  10.36.50              **
**                                                                      **
**************************************************************************
**************************************************************************
```

GEOMETRIC PROPERTIES OF RIVER

+

NO. OF BRANCHES IN UNSTEADY FLOW MODEL = 10
NO. OF GRIDS IN X-DIRCTION = 286
GRID SIZE IN ft. = 500.
NO. OF INTERPOLATIONS BETWN SECTIONS = 3

SECTIONS IN EACH BRANCH

+

BRANCH	SECTIONS INVOLVED FROM	TO
1	1	2
2	3	13
3	14	23
4	24	33
5	34	37
6	38	41
7	42	54
8	55	76
9	77	86
10	87	96

1

INFORMATION ON RIVER SECTIONS

+

SECTION	Lower bank X-CORD	intersection Y-CORD	Angle (rad)	Width (ft.)	Ref datum for depth	No str tubes	Cond. No.	Connect next 1st
1	1657.1	8801.2	1.334	1929	578.90	10	11	2
2	2124.4	8880.5	1.384	1401	577.50	10	11	3
3	2727.6	8955.7	1.479	1019	577.50	10	11	4
4	3535.5	8773.5	1.427	996	577.50	10	11	5
5	4142.4	8645.2	1.338	808	577.20	10	11	6
6	4902.3	8476.0	1.661	979	577.40	10	11	7
7	5756.3	8374.3	1.782	1291	577.20	10	11	8
8	6747.7	8636.6	1.884	1458	577.40	10	11	9
9	8305.2	9623.1	2.075	1442	577.30	10	11	10
10	9054.2	10231.8	2.110	1335	577.20	10	11	11
11	9562.1	10823.5	2.160	1782	577.20	10	11	12
12	10787.6	11865.3	2.164	1719	577.20	10	11	13
13	11230.4	12114.5	2.150	1756	577.20	10	11	14
14	12198.4	12440.6	1.561	2582	578.25	10	11	15
15	12869.2	12384.6	1.502	2601	577.00	10	11	16
16	14185.5	12364.6	1.586	2618	577.00	10	11	17
17	15218.0	12383.8	1.521	2466	576.90	10	11	18
18	16695.8	12201.1	1.334	2061	576.80	10	11	19
19	17659.3	11888.4	1.336	1862	576.70	10	11	20
20	19116.4	11449.2	1.226	1773	578.11	10	11	21
21	20471.3	10940.3	1.189	1806	578.03	10	11	22
22	21707.2	10344.3	1.121	1942	578.01	10	11	23
23	22938.9	9429.6	1.010	2028	577.96	10	11	24
24	24484.8	8328.0	1.035	2103	577.91	10	11	25

25	27014.5	7034.9	1.261	1979	577.73	10	11	26
26	28357.9	6572.7	1.460	2022	577.70	10	11	27
27	30181.4	6104.7	1.401	2191	577.76	10	11	28
28	31991.9	5981.3	1.486	2084	577.67	10	11	29
29	33275.0	6075.3	1.539	1909	577.23	10	11	30
30	35026.1	6050.9	1.560	1952	577.51	10	11	31
31	36112.8	6037.8	1.559	2037	577.46	10	11	32
32	38110.1	5822.8	1.557	2257	577.37	10	11	33
33	41859.1	5235.6	1.758	3506	577.41	10	12	34
34	44008.9	5126.0	1.614	2547	577.35	7	11	35
35	46628.1	5270.0	1.773	1601	577.30	7	11	36
36	49075.2	5670.2	1.832	1644	577.25	7	11	37
37	51118.5	6422.0	1.840	1903	577.20	7	21	42
38	43470.6	8575.7	2.008	1168	577.35	3	11	39
39	45877.8	9273.2	1.683	1169	577.30	3	11	40
40	48178.3	9084.9	1.564	1203	577.25	3	11	41
41	50466.6	8963.7	1.713	1205	577.20	3	21	42
42	53074.5	6858.2	1.726	3608	577.20	10	11	43
43	55524.8	7870.6	1.702	3081	577.26	10	11	44
44	57538.3	8596.4	1.784	2613	577.14	10	11	45
45	58289.1	8843.8	1.713	2385	576.93	10	11	46
46	59233.9	9127.0	1.663	2074	576.93	10	11	47
47	60694.0	9215.6	1.649	2138	576.94	10	11	48
48	61790.0	9223.5	1.617	2352	576.90	10	11	49
49	63184.9	9122.9	1.556	2561	576.83	10	11	50
50	64699.0	8770.7	1.429	3078	576.80	10	11	51
51	66265.6	8390.1	1.396	3392	576.83	10	11	52
52	67608.9	8095.2	1.438	3498	576.75	10	11	53
53	68695.2	8109.0	1.492	3418	576.80	10	11	54
54	69944.2	8199.7	1.566	3338	576.80	10	11	55
55	71124.4	8504.7	1.633	2922	576.75	10	11	56
56	72084.0	8834.2	1.664	2532	576.73	10	11	57
57	73475.3	8991.3	1.689	2302	576.72	10	11	58
58	74713.6	9179.9	1.711	2185	576.71	10	11	59
59	75893.3	9490.0	1.859	2087	576.68	10	11	60
60	77210.9	10295.0	1.991	1704	576.65	10	11	61
61	79155.4	11809.9	2.19[illegible]	[illegible]99	576.66	10	11	62
62	80228.7	12565.7	2.1[illegible]	[illegible]	576.58	10	11	63
63	81605.8	13436.2	2.112	[illegible]	576.58	10	11	64
64	83527.2	14610.7	1.969	[illegible]4	576.50	10	11	65
65	85018.3	15230.4	1.933	1593	576.50	10	11	66
66	86346.9	15929.5	1.912	1666	576.46	10	11	67
67	87968.7	16466.3	1.925	1935	576.43	10	11	68
68	89593.0	17077.4	1.778	1910	576.32	10	11	69
69	91777.0	17153.1	1.596	2376	576.37	10	11	70
70	93564.7	16713.2	1.434	2330	576.35	10	11	71
71	94994.2	16394.8	1.347	2727	576.33	10	11	72
72	97192.3	15533.5	1.381	2819	576.35	10	11	73
73	99502.3	15015.0	1.431	2996	576.12	10	11	74
74	102673.3	15286.4	1.813	2912	576.14	10	11	75
75	104778.3	15863.5	1.828	3100	576.20	10	11	76
76	106070.0	16235.7	1.664	3293	576.27	10	11	77
77	107397.9	16237.6	1.454	3306	575.96	10	11	78
78	111246.2	15533.3	1.478	3935	575.86	10	11	79
79	113869.5	14790.8	1.479	4286	575.86	10	11	80
80	116941.9	14816.9	1.525	2863	575.76	10	11	81
81	118933.9	14945.5	1.515	2823	575.74	10	11	82
82	120905.9	14707.9	1.516	2744	575.70	10	11	83
83	123254.1	14625.2	1.559	2425	575.65	10	11	84
84	125696.6	14531.3	1.605	2349	575.62	10	11	85
85	127595.8	14777.3	1.672	2041	575.58	10	11	86
86	128395.6	14875.7	1.776	2070	575.57	10	11	87
87	129381.7	15216.8	1.797	2008	575.53	10	11	88
88	131323.3	15706.9	1.812	2180	575.51	10	11	89
89	133200.6	16322.5	1.799	2296	575.27	10	11	90
90	134812.4	16907.3	1.776	2269	575.20	10	11	91
91	135705.2	16957.1	1.771	2614	575.21	10	11	92
92	136895.9	17044.1	1.698	2582	575.16	10	11	93
93	137777.1	17058.2	1.617	2246	575.12	10	11	94
94	140042.9	17187.0	1.671	2371	575.12	10	11	95
95	140798.6	17079.9	1.613	2670	575.14	10	11	96
96	141405.6	16845.7	1.342	2931	575.12	10	11	97
97	143773.4	15244.5	1.203	3973	575.04	10	11	98

Geometry of X-Sections

SCTN	Distance and Depth (ft.) in pairs of data							
1	0.: 0.0	212.: 41.6	280.: 41.9	354.: 33.7	906.: 34.5	1230.: 77.8	1377.: 23.3	1625.: 8.4	1929.: 0.0
2	0.: 0.0	185.: 11.0	343.: 38.4	451.: 31.8	600.: 31.5	806.: 39.2	839.: 51.7	998.: 68.6	1041.: 60.3
	1115.: 41.9	1245.: 6.7	1401.: 0.0						
3	0.: 0.0	70.: 24.0	241.: 41.8	345.: 37.6	568.: 66.6	936.: 5.8	1020.: 0.0		
4	0.: 0.0	73.: 7.5	141.: 17.6	157.: 37.5	284.: 65.4	350.: 65.9	420.: 61.0	525.: 60.6	617.: 47.4
	845.: 7.3	996.: 0.0							
5	0.: 0.0	18.: 10.1	76.: 17.7	116.: 33.0	266.: 45.0	362.: 46.0	465.: 56.5	529.: 51.9	619.: 42.1
	710.: 24.1	809.: 0.0							
6	0.: 0.0	12.: 27.6	49.: 35.6	144.: 39.7	186.: 38.4	285.: 43.8	402.: 39.5	470.: 41.8	677.: 41.6
	741.: 37.7	817.: 41.6	980.: 0.0						
7	0.: 0.0	11.: 19.4	56.: 33.4	141.: 43.3	223.: 38.5	310.: 42.6	367.: 42.0	502.: 33.6	575.: 34.0
	820.: 54.5	873.: 54.5	909.: 51.2	977.: 53.3	1169.: 26.5	1227.: 24.2	1292.: 0.0		
8	0.: 0.0	54.: 23.9	135.: 30.8	179.: 47.6	462.: 41.0	590.: 34.5	895.: 51.9	1039.: 43.7	1178.: 30.0
	1413.: 33.5	1458.: 0.0							
9	0.: 0.0	127.: 44.7	384.: 53.4	561.: 49.5	653.: 53.4	798.: 54.4	956.: 43.5	1062.: 25.4	1158.: 26.3
	1276.: 7.7	1443.: 0.0							
10	0.: 0.0	10.: 14.3	155.: 46.4	272.: 43.4	444.: 49.8	537.: 46.4	855.: 46.3	995.: 32.8	1079.: 31.9
	1336.: 0.0								
11	0.: 0.0	84.: 35.1	219.: 40.4	299.: 38.4	524.: 49.3	720.: 44.3	930.: 46.4	1147.: 27.9	1535.: 27.0
	1613.: 19.5	1772.: 9.0	1782.: 0.0						
12	0.: 0.0	183.: 36.1	397.: 33.3	611.: 41.0	781.: 41.9	1312.: 33.0	1567.: 23.3	1719.: 0.0	
13	0.: 0.0	25.: 20.3	146.: 32.3	382.: 36.8	727.: 41.1	1009.: 33.6	1441.: 30.1	1749.: 1.7	1756.: 0.0
14	0.: 0.0	127.: 23.0	639.: 30.0	1203.: 35.8	1432.: 30.8	1730.: 36.5	1956.: 30.4	2383.: 33.1	2486.: 29.6
	2583.: 0.0								
15	0.: 0.0	97.: 26.7	280.: 32.1	420.: 29.1	660.: 36.3	778.: 30.7	1119.: 28.3	1672.: 34.0	1842.: 36.0
	2031.: 31.7	2275.: 46.0	2368.: 19.4	2549.: 14.7	2602.: 0.0				
16	0.: 0.0	180.: 24.8	384.: 33.4	582.: 28.5	720.: 34.7	840.: 29.0	1662.: 29.5	1794.: 26.5	2225.: 39.5
	2419.: 24.2	2531.: 18.6	2618.: 0.0						
17	0.: 0.0	8.: 27.5	90.: 31.5	299.: 27.9	477.: 39.7	619.: 29.1	982.: 36.6	1556.: 33.1	1807.: 37.6
	2123.: 38.1	2375.: 17.8	2466.: 0.0						
18	0.: 0.0	240.: 30.1	430.: 31.9	489.: 37.9	532.: 31.6	883.: 28.9	1134.: 32.5	1345.: 39.9	1763.: 35.8
	1922.: 25.4	2055.: 22.3	2062.: 0.0						
19	0.: 0.0	103.: 8.9	122.: 19.7	241.: 30.7	711.: 32.2	1141.: 28.3	1443.: 32.4	1580.: 27.1	1724.: 28.8
	1776.: 24.2	1853.: 27.1	1863.: 0.0						
20	0.: 0.0	54.: 13.1	292.: 35.4	682.: 30.8	1061.: 39.3	1201.: 36.8	1396.: 47.3	1773.: 0.0	
21	0.: 0.0	237.: 32.5	731.: 38.4	930.: 35.0	1176.: 41.4	1692.: 23.6	1806.: 0.0		
22	0.: 0.0	11.: 10.0	174.: 31.6	1203.: 37.0	1581.: 32.6	1757.: 19.8	1848.: 4.0	1943.: 0.0	
23	0.: 0.0	125.: 25.4	307.: 26.2	673.: 35.2	997.: 32.7	1644.: 36.0	1930.: 19.8	2028.: 0.0	

24	0.: 0.0	0.: 6.7	213.: 26.5	546.: 28.8	1186.: 40.9	1461.: 34.1	1683.: 35.9	1948.: 27.2	2103.: 0.0
25	0.: 0.0	11.: 11.7	102.: 23.4	734.: 38.8	1184.: 43.3	1720.: 30.0	1910.: 5.1	1980.: 0.0	
26	0.: 0.0	60.: 13.4	140.: 30.9	734.: 37.7	885.: 36.9	1048.: 39.4	1864.: 30.6	2023.: 0.0	
27	0.: 0.0	119.: 21.5	577.: 34.7	1173.: 34.4	1737.: 33.5	2045.: 23.8	2192.: 0.0		
28	0.: 0.0	9.: 10.2	329.: 32.0	1083.: 36.1	1897.: 31.9	2085.: 0.0			
29	0.: 0.0	7.: 21.9	257.: 31.6	691.: 31.6	933.: 44.5	1174.: 37.3	1339.: 39.3	1671.: 34.5	1910.: 0.0
30	0.: 0.0	168.: 29.0	1051.: 39.9	1296.: 38.5	1408.: 35.3	1605.: 36.3	1825.: 22.4	1953.: 0.0	
31	0.: 0.0	221.: 31.9	971.: 40.8	1294.: 41.9	1523.: 31.7	1675.: 37.0	2037.: 0.0		
32	0.: 0.0	6.: 3.9	138.: 6.5	320.: 32.8	1033.: 34.2	1877.: 39.5	2135.: 32.7	2166.: 8.6	2257.: 0.0
33	0.: 0.0	39.: 42.5	470.: 31.6	836.: 37.3	912.: 32.7	1114.: 32.3	1753.: 18.0	1870.: 12.0	2026.: 9.0
	2221.: 12.0	2350.: 18.0	2493.: 29.5	2680.: 33.2	2808.: 31.0	2924.: 41.6	3221.: 35.9	3507.: 0.0	
34	0.: 0.0	92.: 7.0	196.: 26.6	564.: 31.6	829.: 28.2	1068.: 30.2	2048.: 10.3	2132.: 3.8	2547.: 0.0
35	0.: 0.0	5.: 6.2	218.: 32.2	314.: 28.6	405.: 31.1	700.: 28.0	920.: 30.8	1142.: 25.6	1236.: 26.2
	1508.: 13.6	1601.: 0.0							
36	0.: 0.0	97.: 26.6	704.: 32.5	927.: 27.2	1033.: 28.1	1183.: 22.7	1537.: 19.2	1644.: 0.0	
37	0.: 0.0	229.: 24.9	657.: 31.1	1034.: 28.9	1174.: 23.8	1608.: 18.7	1643.: 23.1	1734.: 3.6	1903.: 0.0
38	0.: 0.0	258.: 23.9	619.: 26.6	865.: 26.3	910.: 29.4	1053.: 6.6	1169.: 0.0		
39	0.: 0.0	361.: 25.9	507.: 24.8	758.: 31.5	953.: 27.7	1060.: 8.2	1169.: 0.0		
40	0.: 0.0	94.: 3.8	197.: 23.9	319.: 21.9	400.: 24.3	661.: 27.5	898.: 30.5	1069.: 18.1	1138.: 2.4
	1203.: 0.0								
41	0.: 0.0	178.: 4.8	342.: 22.0	560.: 32.5	761.: 29.2	993.: 21.6	1107.: 2.5	1205.: 0.0	
42	0.: 0.0	41.: 3.9	225.: 33.7	1392.: 30.5	1439.: 23.3	2088.: 3.7	2395.: 3.4	2464.: 2.9	2872.: 38.0
	3125.: 30.0	3512.: 22.4	3573.: 5.7	3608.: 0.0					
43	0.: 0.0	35.: 6.0	125.: 32.7	1390.: 32.9	1416.: 24.4	1776.: 22.9	1866.: 23.0	1977.: 16.1	2350.: 43.6
	2624.: 37.5	2745.: 25.1	2963.: 6.5	3082.: 0.0					
44	0.: 0.0	78.: 3.9	264.: 34.8	1470.: 33.0	1729.: 40.2	1973.: 39.1	2206.: 22.1	2340.: 25.2	2505.: 16.8
	2613.: 0.0								
45	0.: 0.0	101.: 4.1	212.: 31.2	389.: 35.5	616.: 31.4	864.: 37.5	1208.: 31.1	1389.: 41.4	1840.: 35.9
	1994.: 25.4	2207.: 17.4	2385.: 0.0						
46	0.: 0.0	206.: 34.6	583.: 32.3	832.: 38.1	1046.: 31.7	1184.: 39.3	1743.: 31.6	2075.: 0.0	
47	0.: 0.0	9.: 2.9	108.: 6.1	201.: 35.9	656.: 31.3	935.: 39.1	1696.: 38.7	1924.: 19.5	2002.: 4.8
	2139.: 0.0								
48	0.: 0.0	130.: 3.4	265.: 33.7	941.: 39.6	1318.: 27.7	1479.: 39.1	2018.: 39.9	2209.: 5.3	2352.: 0.0
49	0.: 0.0	67.: 8.8	174.: 33.1	879.: 36.3	1101.: 27.2	1316.: 26.1	1815.: 38.4	2064.: 39.9	2263.: 27.5
	2353.: 2.9	2555.: 3.5	2561.: 0.0						
50	0.: 0.0	43.: 4.5	166.: 5.7	272.: 32.1	1126.: 34.3	1228.: 18.4	1336.: 25.8	1866.: 26.1	2008.: 34.0
	2706.: 30.3	2078.: 79.0	2831.: 5.7	3061.: 4.4	3078.: 0.0				
51	0.: 0.0	20.: 4.3	110.: 5.4	256.: 32.6	1159.: 35.6	1183.: 19.6	1311.: 10.6	1453.: 18.6	1889.: 28.0
	2836.: 33.2	3127.: 26.6	3222.: 7.1	3393.: 0.0					

52	0.: 0.0	124.: 6.6	208.: 31.2	666.: 32.5	1089.: 32.4	1183.: 10.1	1579.: 9.4	1722.: 23.2	2014.: 28.4
	2398.: 19.3	2791.: 35.3	3111.: 35.9	3357.: 8.4	3498.: 0.0				
53	0.: 0.0	31.: 22.6	439.: 41.4	726.: 31.6	988.: 31.0	1097.: 6.3	1501.: 6.5	1673.: 34.8	2106.: 20.9
	3066.: 35.8	3309.: 7.1	3419.: 0.0						
54	0.: 0.0	113.: 33.4	992.: 31.8	1152.: 5.7	1300.: 40.0	1560.: 33.7	1650.: 22.5	2257.: 21.9	2424.: 29.3
	3050.: 30.0	3218.: 5.7	3339.: 0.0						
55	0.: 0.0	210.: 34.1	661.: 33.8	1131.: 46.1	1311.: 24.2	1757.: 16.9	2221.: 29.5	2771.: 30.0	2922.: 0.0
56	0.: 0.0	47.: 23.4	232.: 31.9	400.: 40.8	520.: 35.3	703.: 33.6	781.: 43.0	1081.: 16.2	1175.: 18.1
	1287.: 13.8	1467.: 22.6	1662.: 23.7	2219.: 28.4	2259.: 27.1	2533.: 0.0			
57	0.: 0.0	95.: 7.5	330.: 48.8	480.: 48.9	880.: 33.9	1216.: 32.4	1259.: 26.6	1593.: 33.1	2091.: 30.5
	2302.: 0.0								
58	0.: 0.0	57.: 25.5	263.: 42.7	480.: 46.2	902.: 33.9	1888.: 30.2	2185.: 0.0		
59	0.: 0.0	77.: 5.0	296.: 36.4	895.: 52.6	997.: 44.2	1117.: 46.0	1331.: 34.7	1736.: 31.4	1911.: 7.1
	2068.: 4.8	2087.: 0.0							
60	0.: 0.0	86.: 26.9	498.: 40.1	984.: 37.8	1181.: 49.7	1551.: 26.6	1637.: 5.6	1701.: 3.3	1704.: 0.0
61	0.: 0.0	1.: 3.1	73.: 4.5	308.: 50.2	436.: 44.4	612.: 49.9	793.: 45.1	1017.: 52.6	1387.: 37.3
	1548.: 5.7	1700.: 0.0							
62	0.: 0.0	338.: 44.5	700.: 45.0	880.: 53.0	1436.: 25.4	1806.: 0.0			
63	0.: 0.0	375.: 40.6	906.: 59.5	1171.: 55.4	1452.: 27.9	1580.: 25.5	1726.: 0.0		
64	0.: 0.0	250.: 46.6	595.: 52.2	1047.: 45.9	1270.: 27.5	1556.: 7.8	1675.: 0.0		
65	0.: 0.0	178.: 21.3	524.: 44.4	828.: 57.2	1281.: 49.2	1450.: 7.0	1594.: 0.0		
66	0.: 0.0	117.: 25.8	739.: 52.8	1278.: 44.8	1448.: 20.7	1532.: 5.7	1633.: 4.4	1667.: 0.0	
67	0.: 0.0	140.: 20.3	354.: 32.7	901.: 47.2	1172.: 49.6	1611.: 34.9	1765.: 11.4	1935.: 0.0	
68	0.: 0.0	0.: 20.1	39.: 27.4	953.: 48.1	1358.: 46.8	1578.: 23.0	1666.: 4.7	1863.: 4.0	1910.: 0.0
69	0.: 0.0	10.: 27.2	608.: 41.9	951.: 43.2	1765.: 30.3	2026.: 5.9	2351.: 4.1	2376.: 0.0	
70	0.: 0.0	125.: 5.3	298.: 31.7	745.: 42.1	973.: 40.2	1286.: 46.9	1660.: 33.5	2050.: 30.1	2230.: 7.1
	2331.: 0.0								
71	0.: 0.0	146.: 4.9	247.: 28.1	628.: 35.8	1090.: 38.0	1433.: 48.8	1750.: 32.9	2094.: 32.1	2339.: 5.8
	2718.: 3.9	2727.: 0.0							
72	0.: 0.0	20.: 3.6	196.: 5.2	297.: 30.6	1061.: 32.6	1604.: 39.5	2228.: 31.8	2476.: 23.8	2535.: 6.0
	2820.: 0.0								
73	0.: 0.0	21.: 5.2	183.: 5.2	324.: 26.8	1295.: 37.1	1878.: 33.7	2193.: 36.3	2665.: 23.5	2714.: 5.6
	2969.: 4.4	2997.: 0.0							
74	0.: 0.0	10.: 2.9	71.: 3.5	254.: 26.6	838.: 31.1	1430.: 33.8	1907.: 46.9	2387.: 27.1	2560.: 28.0
	2721.: 5.7	2904.: 3.3	2912.: 0.0						
75	0.: 0.0	88.: 20.6	319.: 32.1	921.: 35.1	1143.: 45.8	1418.: 42.7	1731.: 28.5	2458.: 24.2	2602.: 5.5
	3101.: 0.0								
76	0.: 0.0	79.: 26.7	492.: 46.3	820.: 43.6	1034.: 32.5	1389.: 25.9	2126.: 24.8	2424.: 26.3	2531.: 7.7
	2809.: 5.4	3067.: 6.9	3291.: 3.7	3293.: 0.0					
77	0.: 0.0	153.: 41.8	289.: 45.1	472.: 38.1	1112.: 32.7	1202.: 23.8	1820.: 19.4	2030.: 24.4	2194.: 37.4
	2406.: 27.7	2550.: 27.0	2677.: 7.6	2858.: 4.5	3071.: 7.8	3241.: 6.6	3307.: 0.0		

```
78     0.:  0.0     117.: 27.0     660.: 33.0    1208.: 27.0    1403.:  6.0    1579.:  6.0    1753.: 18.0    2065.: 18.0    2494.:  2.0
    3000.:  6.0    3460.: 20.0    3740.:  4.0    3935.:  0.0

79     0.:  0.0     312.:  6.0     545.: 27.0    1519.: 27.0    1792.: 29.0    2299.:  0.0    3429.:  0.0    3584.: 27.0    4013.: 26.0
    4167.:  2.0    4286.:  0.0

80     0.:  0.0     236.: 38.4     388.: 33.4     539.: 38.3    1174.: 34.8    1288.: 28.4    1974.: 32.0    2128.:  8.1    2206.: 26.9
    2394.: 45.3    2631.: 31.0    2722.:  6.3    2855.:  5.4    2864.:  0.0

81     0.:  0.0     133.: 31.8     669.: 36.7    1107.: 35.5    1234.: 23.6    1560.: 22.9    1873.: 39.5    2322.: 52.6    2512.: 25.7
    2574.:  4.0    2823.:  0.0

82     0.:  0.0       6.:  5.3     220.: 30.8     666.: 37.3    1235.: 34.1    1304.: 28.0    1438.: 40.9    1700.: 27.4    19'0.: 47.7
    2256.: 42.8    2519.:  5.8    2744.:  0.0

83     0.:  0.0      66.:  5.0     340.: 33.5    1082.: 35.2    1349.: 48.0    1712.: 47.9    1954.: 30.2    2201.: 22.7    2302.:  6.1
    2426.:  0.0

84     0.·  0.0      64.:  5.6     169.:  6.5     404.: 26.9     910.: 41.6    1013.. 52.3    1428.: 52.6    194 .: 27.9    2212.:  7.7

    2350.:  0.0

85     0.:  0.0      69.:  5.0     136.: 18.2     359.: 30.9     596.: 33.4    1138.: 59.2    1526.: 56.4    2042.:  0.0

86     0.:  0.0       7.:  3.3     161.:  4.5     329.: 33.4     717.: 34.7     992.: 53.2    1318.: 46.3    1534.: 55.7    1810.: 40.8
    1928.:  7.8    2071.:  0.0

87     0.:  0.0       2.:  5.1      49.:  5.6     110.: 17.9     309.: 34.2     587.: 34.8     775.: 40.3     978.: 53.1    1289.: 43.8
    1531.: 47.9    1782.: 26.4    1879.:  5.3    2008.:  0.0

88     0.:  0.0      97.:  5.2     193.: 22.4     397.: 33.6     686.: 33.1     883.: 38.3    1092.: 51.0    1369.: 47.4    1782.: 49.0
    2072.:  6.7    2181.:  0.0

89     0.:  0.0     215.: 23.9     446.: 33.0     852.: 35.6    1345.: 53.2    1764.: 50.7    2098.:  4.5    2296.:  0.0

90     0.:  0.0      98.:  4.5     225.: 42.1     459.: 62.2     822.: 44.4    1598.: 33.9    1941.:  6.6    2270.:  0.0

91     0.:  0.0      50.:  5.3     239.: 51.0     417.: 54.5    1551.: 31.3    2049.: 27.8    2137.:  5.7    2615.:  0.0

92     0.:  0.0     175.: 46.3     660.: 47.0    1554.: 25.3    2044.: 21.9    2180.:  5.7    2583.:  0.0

93     0.:  0.0      83.:  3.5     217.: 31.6     386.: 37.8     702.: 44.9     876.: 56.9    1051.: 49.0    1348.: 46.5    1707.: 26.5
    1919.: 27.4    2108.:  5.4    2246.:  0.0

94     0.:  0.0     255.: 33.4     787.: 33.9    1272.: 47.3    1574.: 43.5    1758.: 35.8    1988.: 33.1    2090.:  6.6    2372.:  0.0

95     0.:  0.0      49.: 10.3     403.: 32.7     884.: 43.5    1581.: 40.6    1702.: 35.2    1990.: 38.3    2250.:  5.7    2671.:  0.0

96     0.:  0.0       1.:  5.1     105.:  4.4     268.: 40.8     594.: 39.3    1211.: 32.5    1459.: 37.2    1697.: 32.4    2152.: 46.3
    2394.: 28.6    2491.:  5.1    2932.:  0.0

97     0.:  0.0      31.: 25.2     131.: 40.1     569.: 46.3     914.: 33.5    1125.: 30.9    1528.: 10.7    2138.:  5.1    2356.:  6.0
    2520.: 16.4    2547.: 33.3    2831.: 41.5    3240.: 36.4    3556.: 35.9    3681.: 14.8    3963.: 10.8    3973.:  0.0
1
```

GRID CONFIGURATION and BOUNDARY TYPES OF SCHEMATIZED RIVER

X GRID	Y GRID OF Bank 1	Bank 2	Bank 3	Bank 4	REJECTION RATE PER TIME STEP Bank 1	Bank 2	Bank 3	Bank 4
1	17	28	0	0	.1591	.1591		
2	18	26	0	0	.1591	.1591		
3	18	24	0	0	.1591	.1591		
4	19	23	0	0	.1591	.0000		
5	19	21	0	0	.1591	.9948		
6	19	20	0	0	.0000	.9948		
7	19	20	0	0	.0000	.9948		
8	18	19	0	0	.9948	.9948		
9	18	19	0	0	.9948	.9948		
10	18	19	0	0	.9948	.9948		
11	18	19	0	0	.9948	.9948		

12	18	20	0	0	.9948	.9948		
13	18	20	0	0	.9948	.9948		
14	18	20	0	0	.9948	.9948		
15	19	21	0	0	.9948	.9948		
16	19	22	0	0	.9948	.9948		
17	20	24	0	0	.9948	.9948		
18	21	26	0	0	.9948	.0000		
19	22	30	27	28	.9948	.9948	.0	.0
20	23	30	27	27	.9948	.9948	.0	.0
21	24	30	0	0	.9948	.0000		
22	25	30	0	0	.9948	.0000		
23	25	30	0	0	.9948	.0000		
24	26	30	0	0	.9948	.9948		
25	26	30	0	0	.9948	.9948		
26	26	30	0	0	.9948	.9948		
27	26	30	0	0	.9948	.9948		
28	26	30	0	0	.1591	.9948		
29	26	30	0	0	.1591	.9948		
30	26	30	0	0	.1591	.9948		
31	26	29	0	0	.1591	.9948		
32	26	29	0	0	.1591	.9948		
33	26	29	0	0	.1591	.9948		
34	25	28	0	0	.1591	.9948		
35	25	28	0	0	.1591	.9948		
36	25	28	0	0	.1591	.9948		
37	24	27	0	0	.1591	.9948		
38	24	27	0	0	.1591	.1591		
39	24	26	0	0	.1591	.1591		
40	23	26	0	0	.1591	.1591		
41	23	26	0	0	.1591	.1591		
42	23	26	0	0	.0072	.1591		
43	22	25	0	0	.0072	.0000		
44	22	25	0	0	.0072	.0000		
45	21	24	0	0	.0072	.0000		
46	20	24	0	0	.0000	.0000		
47	19	23	0	0	.0000	.0000		
48	19	23	0	0	.0000	.0000		
49	18	22	0	0	.0072	.9948		
50	17	21	0	0	.0072	.9948		
51	17	20	0	0	.0072	.9948		
52	16	20	0	0	.1591	.9948		
53	16	19	0	0	.1591	.9948		
54	15	19	0	0	.1591	.0000		
55	15	18	0	0	.1591	.9948		
56	15	18	0	0	.1591	.9948		
57	14	17	0	0	.1591	.9948		
58	14	17	0	0	.1591	.1591		
59	14	17	0	0	.1591	.1591		
60	13	17	0	0	.1591	.1591		
61	13	17	0	0	.1591	.1591		
62	13	16	0	0	.1591	.1591		
63	13	16	0	0	.1591	.1591		
64	13	16	0	0	.1591	.1591		
65	13	16	0	0	.1591	.1591		
66	13	16	0	0	.9948	.1591		
67	13	16	0	0	.9948	.1591		
68	13	16	0	0	.9948	.1591		
69	13	16	0	0	.9948	.1591		
70	13	16	0	0	.9948	.1591		
71	13	16	0	0	.9948	.1591		
72	13	16	0	0	.9948	.1591		
73	13	16	0	0	.9948	.1591		
74	13	16	0	0	.9948	.1591		
75	13	16	0	0	.0072	.1591		
76	13	16	0	0	.0072	.1591		
77	13	16	0	0	.0072	.0072		
78	13	16	0	0	.0072	.0072		
79	13	16	0	0	.0072	.0072		
80	12	16	0	0	.0000	.0072		
81	12	17	0	0	.1591	.1591		
82	12	17	0	0	.1591	.1591		
83	12	17	0	0	.1591	.1591		

84	12	18	0	0	.1591	.1591		
85	12	18	0	0	.9948	.1591		
86	11	19	0	0	.9948	.1591		
87	11	19	16	17	.9948	.1591	.1591	.1591
88	11	20	16	17	.9948	.0072	.1591	.0072
89	11	20	16	18	.9948	.0072	.0072	.0072
90	11	21	15	18	.1591	.0072	.0072	.0072
91	11	21	15	18	.1591	.0072	.0072	.0072
92	11	21	15	19	.0072	.0072	.0072	.0072
93	11	21	15	19	.0072	.0072	.0072	.0072
94	12	21	15	19	.0072	.0072	.1591	.0072
95	12	21	15	18	.0072	.0072	.1591	.0072
96	12	20	16	18	.1591	.0072	.1591	.0072
97	12	20	16	18	.1591	.1591	.1591	.0072
98	13	20	16	18	.1591	.1591	.1591	.0072
99	13	20	16	18	.1591	.1591	.1591	.0072
100	13	20	17	18	.1591	.1591	.1591	.0072
101	13	20	17	18	.1591	.1591	.1591	.1591
102	14	20	18	18	.0072	.1591	.1591	.1591
103	14	21	18	18	.0072	.1591	.1591	.1591
104	14	21	18	18	.0072	.0072	.1591	.1591
105	14	21	18	18	.0072	.0072	.1591	.1591
106	14	21	0	0	.0072	.0072		
107	15	21	0	0	.0072	.0072		
108	15	21	0	0	.0072	.0072		
109	16	21	0	0	.9948	.0072		
110	16	21	0	0	.9948	.0072		
111	16	22	0	0	.0072	.0072		
112	17	22	0	0	.0072	.0072		
113	17	22	0	0	.0072	.0072		
114	18	22	0	0	.0072	.0072		
115	18	22	0	0	.0072	.0072		
116	18	22	0	0	.0072	.1591		
117	19	22	0	0	.0072	.1591		
118	19	22	0	0	.0072	.1591		
119	19	22	0	0	.0072	.1591		
120	19	22	0	0	.0072	.1591		
121	19	22	0	0	.0072	.1591		
122	19	23	0	0	.0072	.0072		
123	19	23	0	0	.0072	.0072		
124	19	23	0	0	.0072	.0072		
125	19	23	0	0	.0072	.0072		
126	19	23	0	0	.0072	.0000		
127	19	23	0	0	.0072	.0000		
128	19	24	0	0	.0072	.0000		
129	19	24	0	0	.0072	.0000		
130	19	24	0	0	.0072	.1591		
131	18	23	0	0	.0072	.1591		
132	18	23	0	0	.0072	.1591		
133	18	23	0	0	.0072	.1591		
134	18	23	0	0	.0072	.1591		
135	17	23	0	0	.0072	.1591		
136	17	23	0	0	.0072	.1591		
137	17	23	0	0	.0072	.1591		
138	17	23	0	0	.9948	.1591		
139	17	23	0	0	.9948	.1591		
140	17	23	0	0	.9948	.1591		
141	17	23	0	0	.9948	.1591		
142	18	23	0	0	.9948	.0000		
143	18	23	0	0	.9948	.9948		
144	18	23	0	0	.9948	.9948		
145	19	23	0	0	.9948	.9948		
146	19	23	0	0	.9948	.9948		
147	19	23	0	0	.9948	.9948		
148	19	23	0	0	.0072	.1591		
149	19	23	0	0	.0072	.1591		
150	19	23	0	0	.0072	.1591		
151	20	23	0	0	.0072	.1591		
152	20	23	0	0	.0072	.1591		
153	20	24	0	0	.0072	.1591		
154	21	24	0	0	.0072	.1591		
155	22	25	0	0	.0072	.1591		

156	22	25	0	0	.9948	.1591		
157	23	26	0	0	.9948	.1591		
158	24	27	0	0	.9948	.1591		
159	25	28	0	0	.9948	.9948		
160	25	29	0	0	.9948	.9948		
161	26	29	0	0	.9948	.9948		
162	27	30	0	0	.9948	.9948		
163	27	30	0	0	.0072	.9948		
164	28	31	0	0	.0072	.9948		
165	29	32	0	0	.0072	.9948		
166	29	32	0	0	.0072	.9948		
167	30	33	0	0	.0072	.9948		
168	31	33	0	0	.0072	.9948		
169	31	33	0	0	.0072	.9948		
170	31	34	0	0	.0072	.9948		
171	32	34	0	0	.0072	.9948		
172	32	35	0	0	.0072	.9948		
173	33	35	0	0	.0072	.9948		
174	33	36	0	0	.0072	.9948		
175	34	36	0	0	.0072	.9948		
176	34	37	0	0	.0072	.9948		
177	34	37	0	0	.0072	.9948		
178	34	37	0	0	.0072	.9948		
179	35	38	0	0	.0072	.1591		
180	35	38	0	0	.9948	.1591		
181	35	38	0	0	.9948	.1591		
182	35	39	0	0	.9948	.1591		
183	35	39	0	0	.9948	.1591		
184	35	39	0	0	.0072	.1591		
185	35	39	0	0	.0072	.1591		
186	35	39	0	0	.0072	.1591		
187	35	39	0	0	.0072	.1591		
188	34	39	0	0	.0072	.1591		
189	34	39	0	0	.0072	.1591		
190	34	39	0	0	.0072	.1591		
191	34	38	0	0	.0072	.1591		
192	33	38	0	0	.0072	.1591		
193	33	38	0	0	.0072	.1591		
194	33	37	0	0	.0072	.1591		
195	32	37	0	0	.0072	.1591		
196	32	37	0	0	.0072	.1591		
197	32	37	0	0	.0072	.1591		
198	31	36	0	0	.1591	.1591		
199	31	36	0	0	.1591	.1591		
200	31	36	0	0	.1591	.1591		
201	31	36	0	0	.1591	.1591		
202	31	36	0	0	.1591	.1591		
203	31	36	0	0	.1591	.1591		
204	31	36	0	0	.9948	.1591		
205	31	36	0	0	.9948	.1591		
206	32	37	0	0	.9948	.1591		
207	32	37	0	0	.9948	.1591		
208	32	37	0	0	.9948	.1591		
209	32	38	0	0	.9948	.1591		
210	33	38	0	0	.9948	.1591		
211	33	39	0	0	.9948	.1591		
212	33	39	0	0	.9948	.1591		
213	33	39	0	0	.9948	.1591		
214	33	39	0	0	.9948	.1591		
215	33	39	0	0	.9948	.1591		
216	33	39	0	0	.9948	.1591		
217	33	39	0	0	.9948	.1591		
218	33	39	0	0	.9948	.1591		
219	33	39	0	0	.9948	.1591		
220	33	39	0	0	.9948	.1591		
221	32	39	0	0	.9948	.1591		
222	32	39	0	0	.9948	.1591		
223	32	39	0	0	.9948	.1591		
224	32	39	0	0	.9948	.1591		
225	32	39	0	0	.9948	.9948		
226	32	39	37	37	.9948	.9948	.1591	.9948
227	31	39	36	37	.9948	.9948	.1591	.9948

228	31	39	35	37	.0000	.9948	.1591	.9948
229	30	38	35	36	.0000	.9948	.1591	.9948
230	30	38	35	36	.0000	.9948	.1591	.9948
231	30	37	0	0	.0000	.9948		
232	30	36	0	0	.0000	.0072		
233	31	36	0	0	.0000	.0072		
234	31	36	0	0	.0000	.0072		
235	31	36	0	0	.0000	.0072		
236	31	36	0	0	.0000	.0072		
237	31	36	0	0	.0000	.0072		
238	31	36	0	0	.0000	.0072		
239	31	35	0	0	.0072	.0072		
240	31	35	0	0	.0072	.0072		
241	30	35	0	0	.0072	.0072		
242	30	35	0	0	.0072	.0072		
243	30	35	0	0	.0072	.1591		
244	30	35	0	0	.0072	.1591		
245	30	35	0	0	.0072	.1591		
246	30	34	0	0	.0072	.1591		
247	30	34	0	0	.0072	.1591		
248	30	34	0	0	.0072	.1591		
249	30	34	0	0	.0072	.1591		
250	30	34	0	0	.0072	.1591		
251	30	34	0	0	.0072	.1591		
252	30	34	0	0	.0072	.1591		
253	30	34	0	0	.0072	.1591		
254	30	34	0	0	.0072	.1591		
255	31	34	0	0	.0072	.1591		
256	31	34	0	0	.0072	.1591		
257	31	34	0	0	.0072	.0072		
258	31	34	0	0	.0072	.0072		
259	31	34	0	0	.0072	.0072		
260	31	35	0	0	.0072	.0072		
261	32	35	0	0	.9948	.0072		
262	32	35	0	0	.9948	.0072		
263	32	36	0	0	.9948	.0072		
264	33	36	0	0	.9948	.0072		
265	33	37	0	0	.9948	.0072		
266	33	37	0	0	.9948	.0072		
267	34	38	0	0	.9948	.0072		
268	34	38	0	0	.9948	.0072		
269	34	38	0	0	.9948	.0072		
270	35	38	0	0	.9948	.0072		
271	35	39	0	0	.0072	.0072		
272	35	39	0	0	.0072	.0072		
273	35	40	0	0	.0072	.0072		
274	35	39	0	0	.0072	.1591		
275	35	39	0	0	.0072	.1591		
276	35	39	0	0	.0072	.1591		
277	35	39	0	0	.0072	.1591		
278	35	39	0	0	.0072	.1591		
279	35	39	0	0	.0072	.1591		
280	35	39	0	0	.0072	.1591		
281	35	39	0	0	.0072	.1591		
282	35	40	0	0	.0072	.1591		
283	35	40	0	0	.0072	.1591		
284	34	40	0	0	.0072	.1591		
285	34	39	0	0	.9948	.1591		
286	33	39	0	0	.9948	.1591		

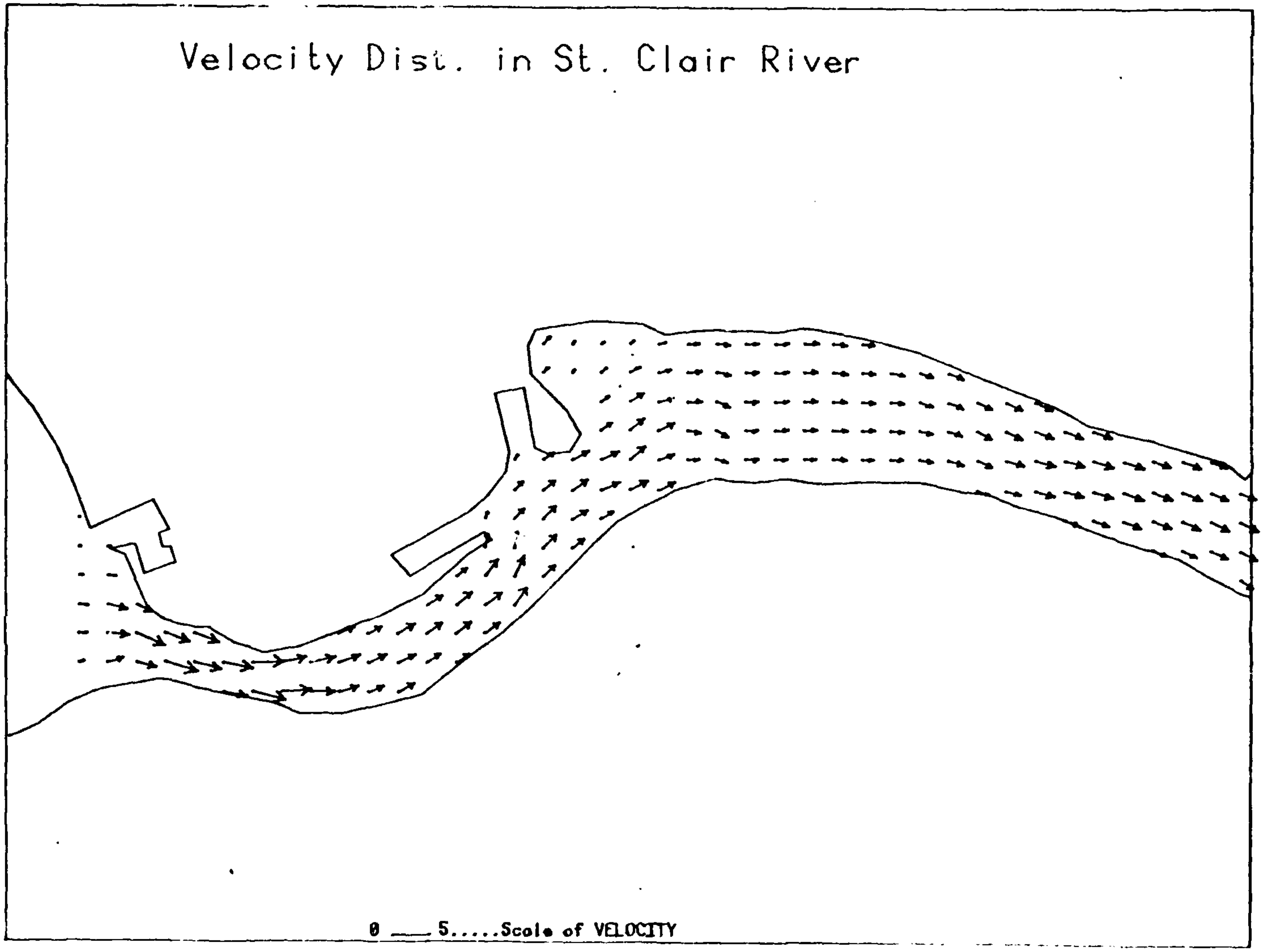


Fig. 11 Velocity Distribution in St. Clair River

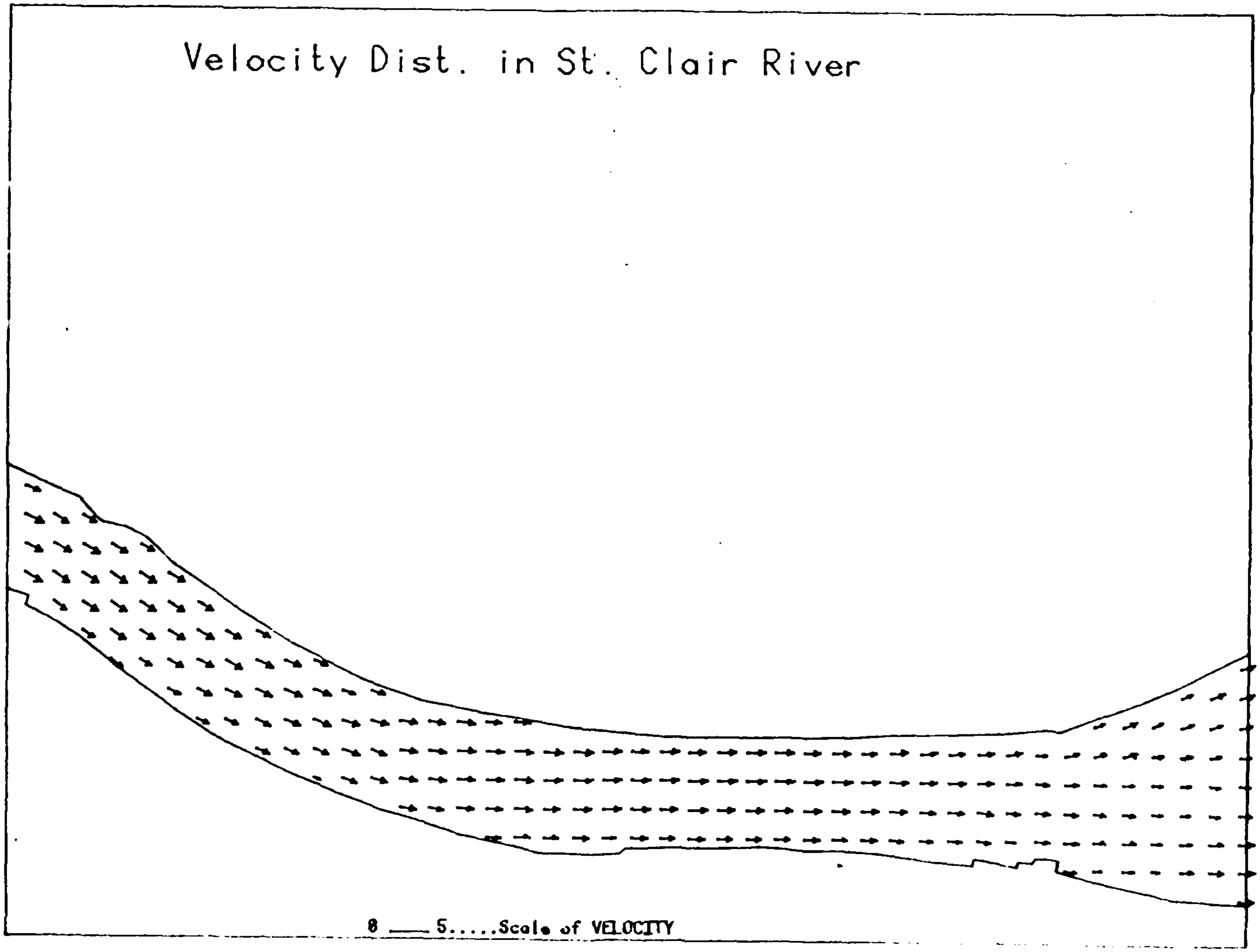


Fig. 11. Velocity Distribution in St. Clair River

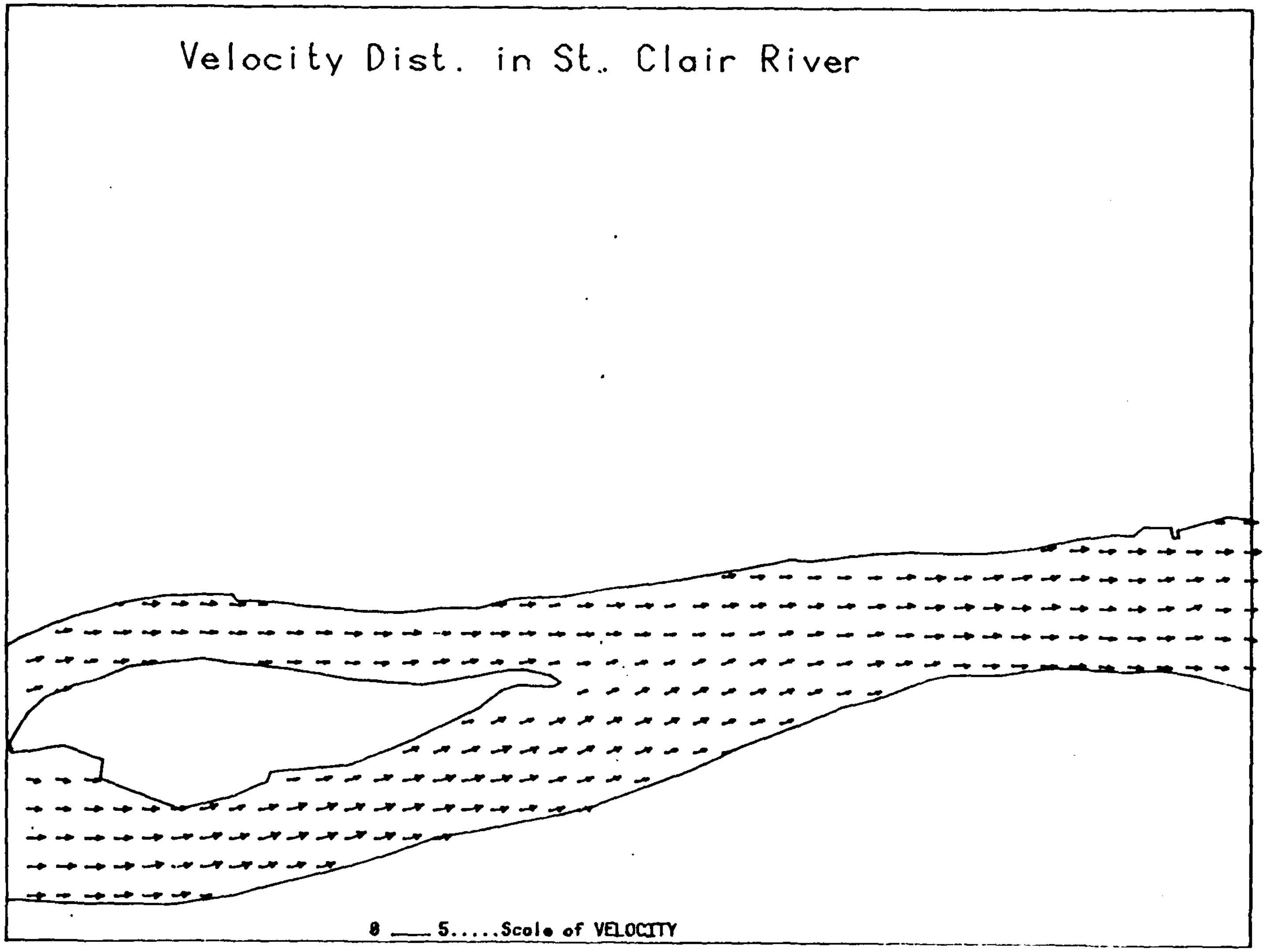


Fig. 11. Velocity Distribution in St. Clair River

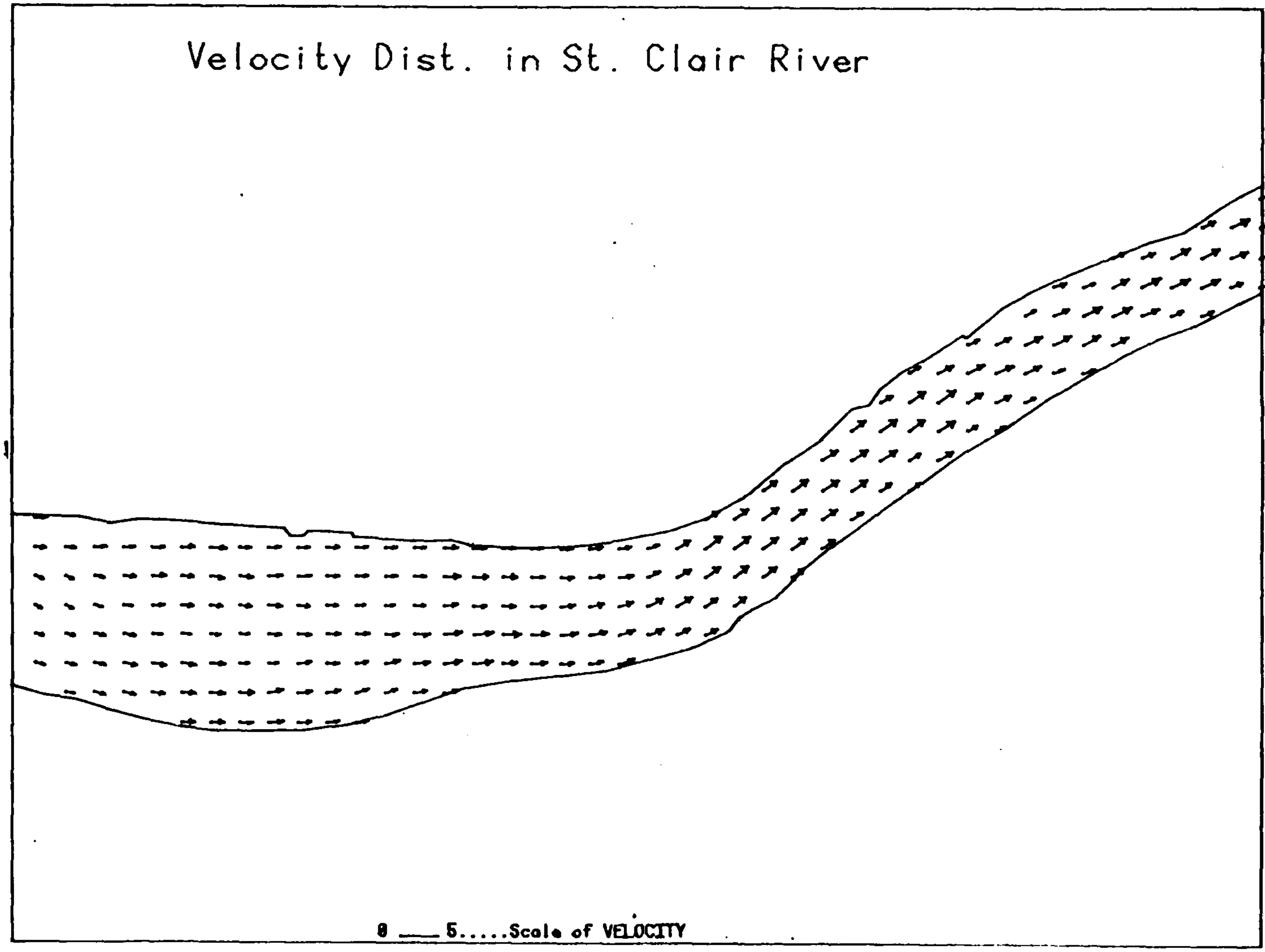


Fig. 11. Velocity Distribution in St. Clair River

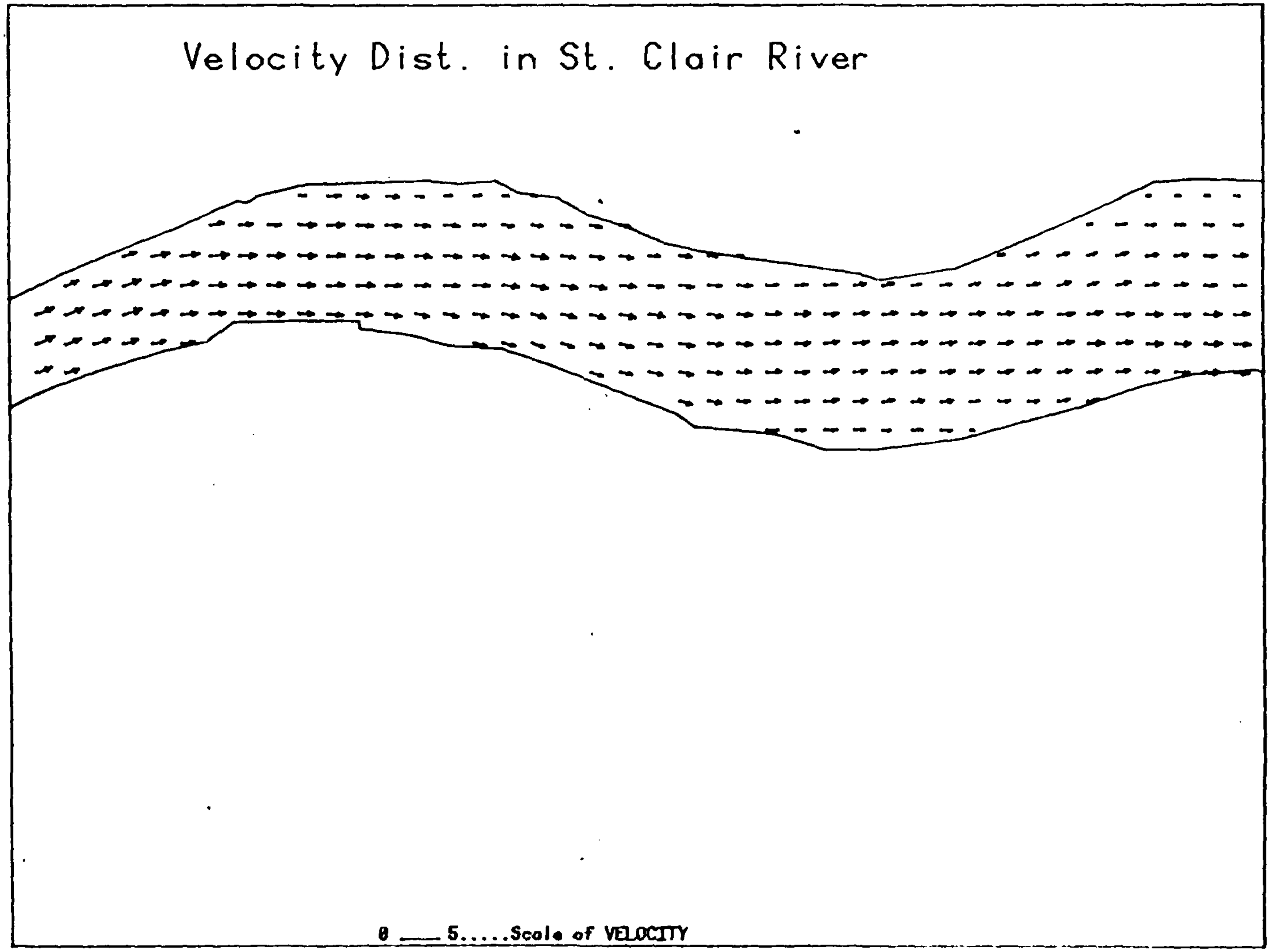


Fig. 11. Velocity Distribution in St. Clair River

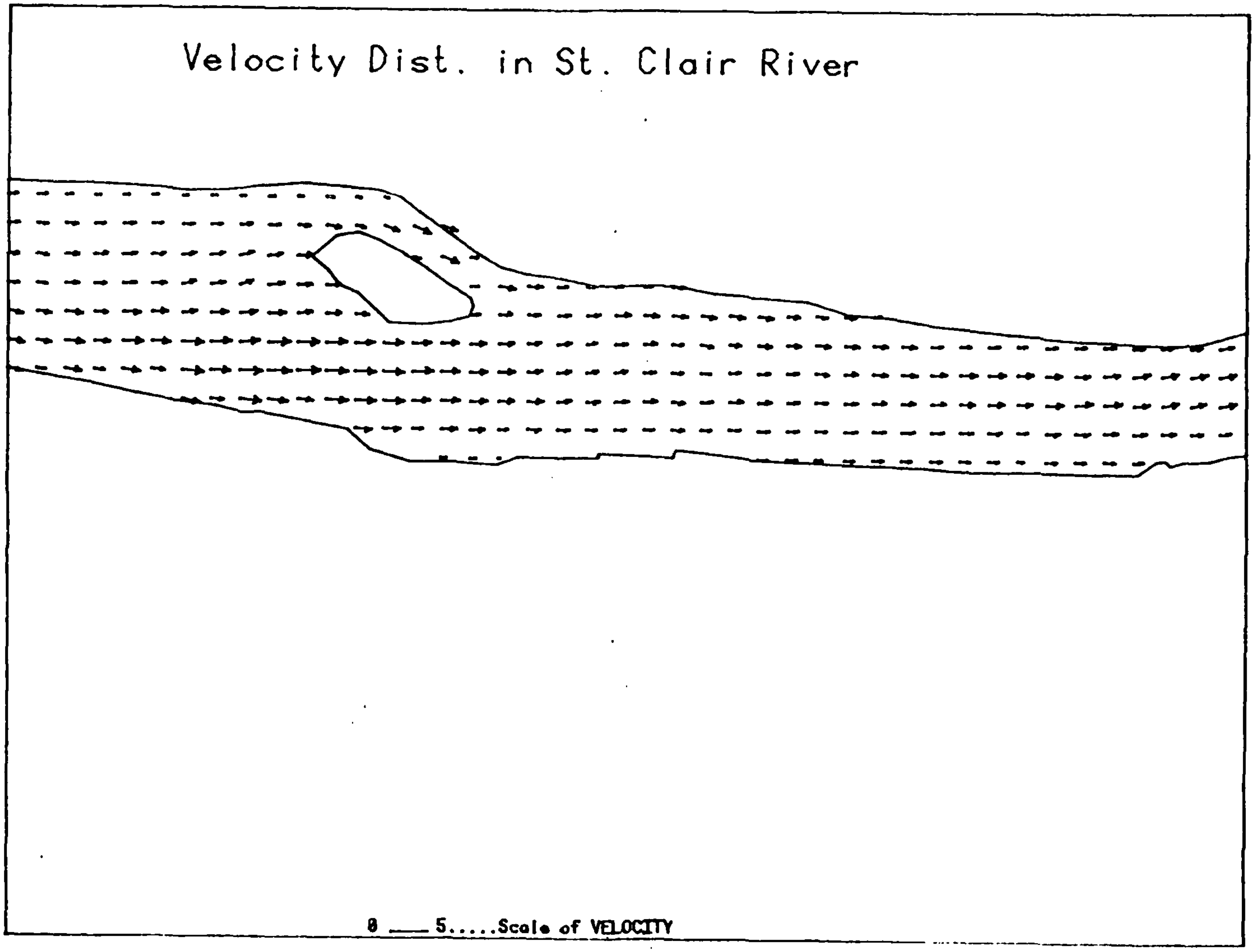


Fig. 11. Velocity Distribution in St. Clair River

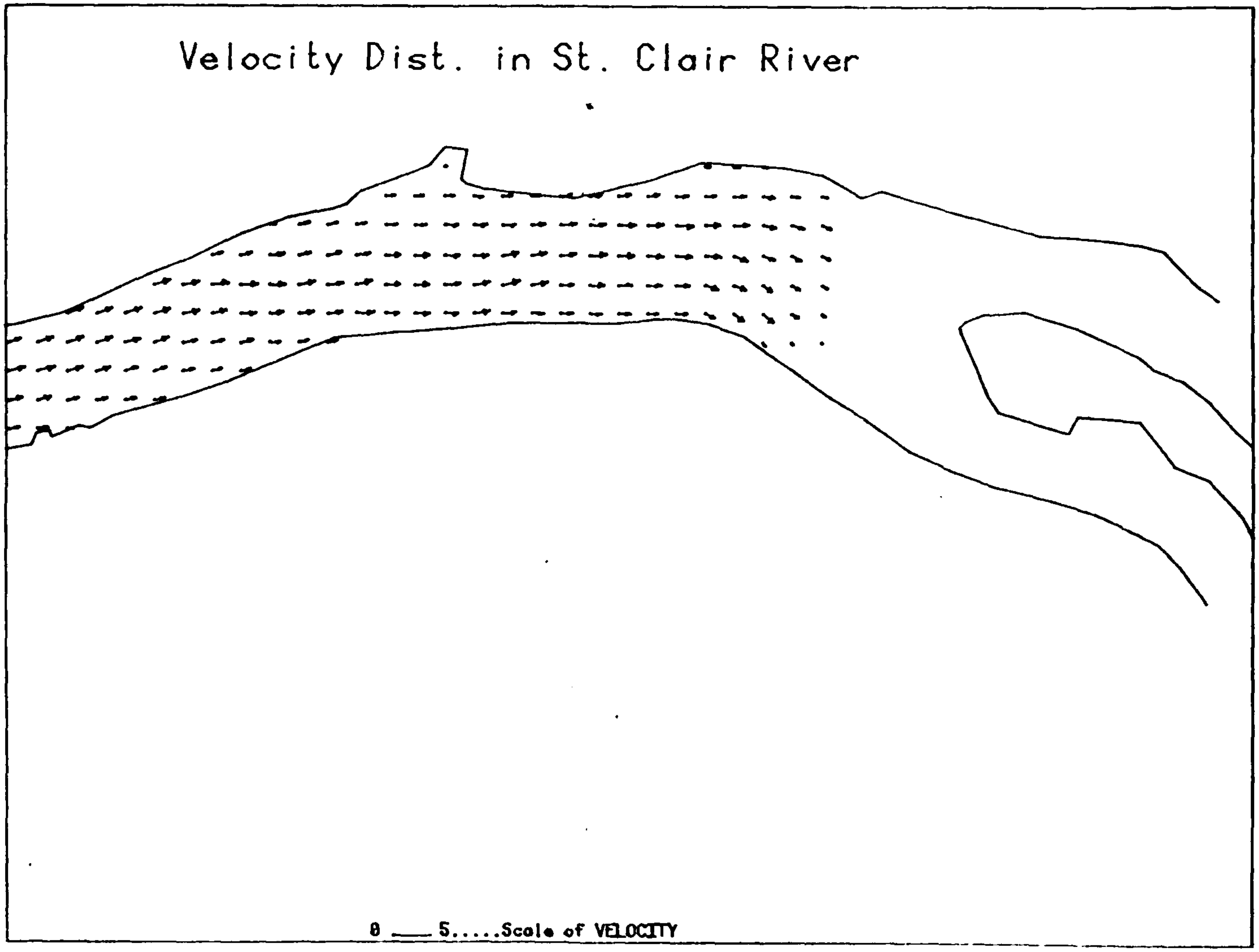


Fig. 11. Velocity Distribution in St. Clair River

St. Clair River

```
*************************
*  INSTANTANEOUS SPILL  *
*          AT           *
*    71750.,  9250.     *
*************************
```

SIMULATION PERIOD = 2.0 Hrs

Characteristics of spill

No. of particles	: 1000
Oil spilled	: 5000. gals of No. 2 Fuel Oil
DT for spill simulation	: 900. Secs.
Specific gravity of oil	: 0.84 (API index = 37.0)
Kinematic Visco. of Water	:0.1411E-04 sq ft/sec
Surafce Tension	:0.2060E-02 lbs/ft

Spreading Coefficients

K2i	K2v	K2t	c10	c20	c30
1.14	0.98	1.60	1.39	1.39	1.43

Molar volume	:0.7063E-02 cu ft/mol
Solubility of fresh oil	:0.1873E-02 lbs/cu ft
Viscosity of Oil	: 0.84lbs/ft-sec
Manning s Roughness of Ice	: 0.035

API option is not selected . Evap. constants are C = 7.88 TO = 465.0

Surface Diffusion - Default formulation is used

Time step for river flow computation = 2.00 Hrs

Open Water Conditions

Flow and Discharge Conditions in River

Branch	Q (cfs)	Stage (ft)
1	129980.	574.99
2	129980.	574.85
3	129970.	574.54
4	129950.	574.28
5	90916.	573.91
6	38964.	573.91
7	129810.	573.60
8	129880.	573.18
9	129810.	572.66
10	129780.	572.35

Time = 0.25 Hrs -- Wind :mag= 2.0 mph, dir =270.0 deg -- Air Temp= 70.0 F
Spill center after advection= 74022., 9638.(ft)
Volume per particle = 5.00 gals

Slick Condition during this time step

Slick information by pie / strip

Pie	No. of particles	Mean radius(ft)
1	175	146.
2	138	140.
3	82	130.
4	87	134.
5	146	156.
6	139	143.
7	103	127.
8	102	124.

Slick condition at the end of this time step

Fraction Evaporated = .52053E-02
Amount Dissolved (gals) : This Step = .64750E-01 Total = .64750E-01

Time = 0.50 Hrs -- Wind :mag= 2.0 mph, dir =270.0 deg -- Air Temp= 70.0 F
Spill center after advection= 76142., 10572.(ft)
Volume per particle = 4.97 gals

Slick Condition during this time step

Slick information by pie / strip

Pie	No. of particles	Mean radius(ft)
1	148	253.
2	218	204.
3	102	226.
4	58	235.
5	114	265.
6	179	287.
7	80	218.
8	66	201.

Slick condition at the end of this time step

Fraction Evaporated = .24043E-01
Amount Dissolved (gals) : This Step = .25054 Total = .31529

Time = 0.75 Hrs -- Wind :mag= 2.0 mph, dir =270.0 deg -- Air Temp= 70.0 F
Spill center after advection= 78110., 12100.(ft)
Volume per particle = 4.88 gals

Slick Condition during this time step

Slick information by pie / strip

Pie	No. of particles	Mean radius(ft)
1	190	356.
2	248	355.
3	82	317.
4	48	296.
5	107	356.
6	214	432.
7	50	284.
8	34	231.

Oil in River Banks

Bank 2; X-Grid 156
Particles 7

Slick condition at the end of this time step

Fraction Evaporated = .58143E-01
Amount Dissolved (gals) : This Step = .54172 Total = .85701

Time = 1.00 Hrs -- Wind :mag= 2.0 mph, dir =270.0 deg -- Air Temp= 70.0 F
Spill center after advection= 80173., 13639.(ft)
Volume per particle = 4.71 gals

Slick Condition during this time step

Slick information by pie / strip

Pie	No. of particles	Mean radius(ft)
1	220	495.
2	209	469.
3	73	354.
4	60	380.
5	116	470.
6	235	565.
7	23	311.
8	35	317.

Oil in River Banks

Bank 2; X-Grid 156 157
Particles 11 2

Slick condition at the end of this time step

Fraction Evaporated = .10265
Amount Dissolved (gals) : This Step = .93088 Total = 1.7879

```
--------------------------------------------------------------------------------
Time =   1.25 Hrs -- Wind :mag=  2.0 mph, dir =270.0 deg   -- Air Temp= 70.0 F
Spill center after advection=  82288., 14898.(ft)
Volume per particle         =     4.48 gals

     Slick Condition during this time step

Slick information by pie / strip
Pie  No. of particles  Mean radius(ft)
 1        282            609.
 2        151            445.
 3         64            358.
 4         77            550.
 5        125            570.
 6        162            611.
 7         30            321.
 8         36            339.

                    Oil in River Banks

Bank 2; X-Grid 156 157 161 163
    Particles  11   2   1  10


Slick condition at the end of this time step

Fraction Evaporated = .14809
Amount Dissolved (gals)      : This Step = 1.3037     Total = 3.0916
```

```
Time =   1.50 Hrs -- Wind :mag=  2.0 mph, dir =270.0 deg   -- Air Temp= 70.0 F
Spill center after advection=  84487., 16099.(ft)
Volume per particle          =     4.26 gals

     Slick Condition during this time step

Slick information by pie / strip
Pie  No. of particles  Mean radius(ft)
 1        321              811.
 2        105              523.
 3         36              372.
 4         98              505.
 5        163              851.
 6        131              750.
 7         27              339.
 8         40              466.

                          Oil in River Banks

Bank 1; X-Grid 168
     Particles   8

Bank 2; X-Grid 156 157 161 163 169
     Particles  11   2   1  16  14


Slick condition at the end of this time step

Fraction Evaporated = .19416
Amount Dissolved (gals)      : This Step = 1.8238     Total = 4.9154
```

Time = 1.75 Hrs -- Wind :mag= 2.0 mph, dir =270.0 deg -- Air Temp= 70.0 F
Spill center after advection= 86822., 17160.(ft)
Volume per particle = 4.02 gals

Slick Condition during this time step

Slick information by pie / strip

Strip	Particles	-Le(ft)	Y mean	Le(ft)
167	2	-72.	15016.	72.
168	4	-73.	15032.	73.
171	41	-166.	16035.	265.
172	71	-355.	16546.	603.
173	122	-381.	16896.	371.
174	191	-347.	17200.	305.
175	231	-276.	17445.	259.
176	139	-205.	17559.	180.

Oil in River Banks

Bank 1; X-Grid 168 175
Particles 10 2

Bank 2; X-Grid 156 157 161 163 169 171 173 175
Particles 11 2 1 16 72 57 27 5

Slick condition at the end of this time step

Fraction Evaporated = .24656
Amount Dissolved (gals) : This Step = 2.9435 Total = 7.8589

```
---------------------------------------------------------------------------
Time =   2.00 Hrs -- Wind :mag=  2.0 mph, dir =270.0 deg   -- Air Temp= 70.0 F
Spill center after advection=  89002., 17805.(ft)
Volume per particle          =     3.76 gals

      Slick Condition during this time step

Slick information by pie / strip
Strip  Particles  -Le(ft)   Y mean   Le(ft)
173          6      -89.   17438.     89.
176         59     -235.   17167.    404.
177         79     -340.   17558.    520.
178        141     -359.   17800.    355.
179        172     -331.   17980.    285.
180        197     -297.   18076.    255.
181         75     -254.   18116.    212.

                        Oil in River Banks

Bank 1; X-Grid 168 175
    Particles   9   6

Bank 2; X-Grid 156 157 161 163 169 171 173 175 178
    Particles  11   2   1  16  61  52  44  17   3


Slick condition at the end of this time step

Fraction Evaporated = .29250
Amount Dissolved (gals)      : This Step = 3.6486     Total = 11.507
```

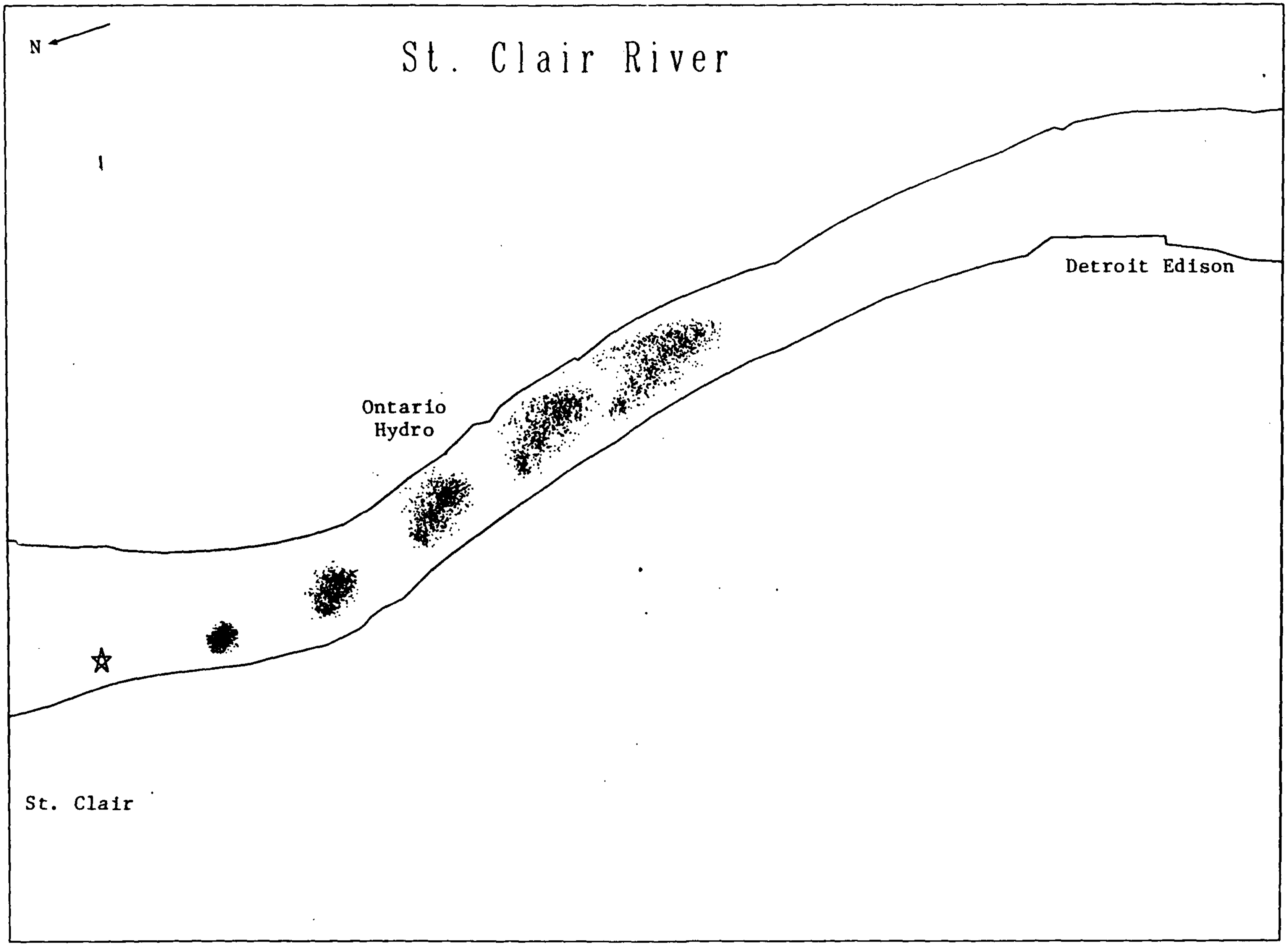


Fig. 12. Plot of slick locations at t = 15 min, 30 min, 45 min, 1 hr and 1 hr 15 min

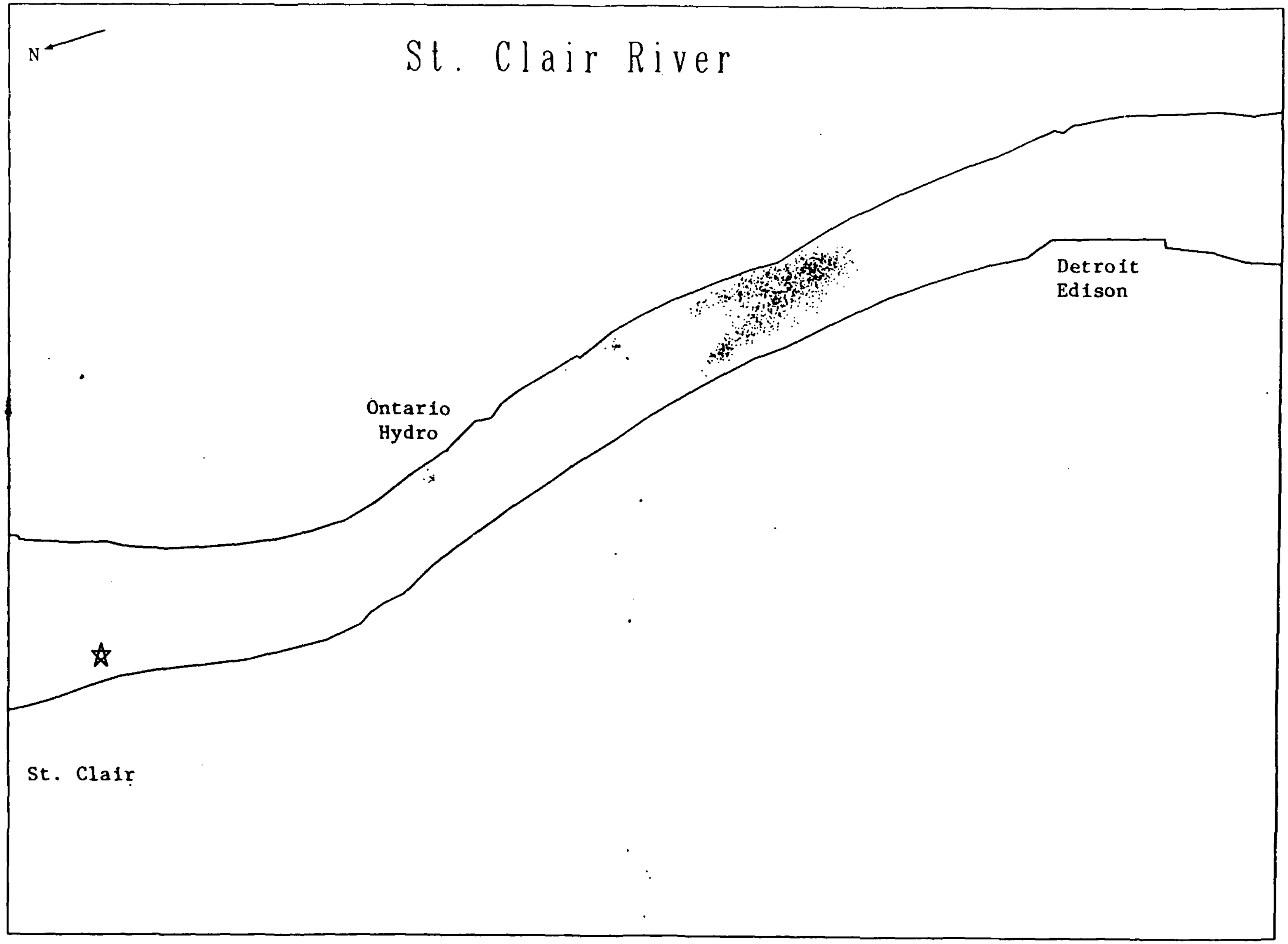


Fig. 13. Plot of slick location at t = 1 hr 30 min

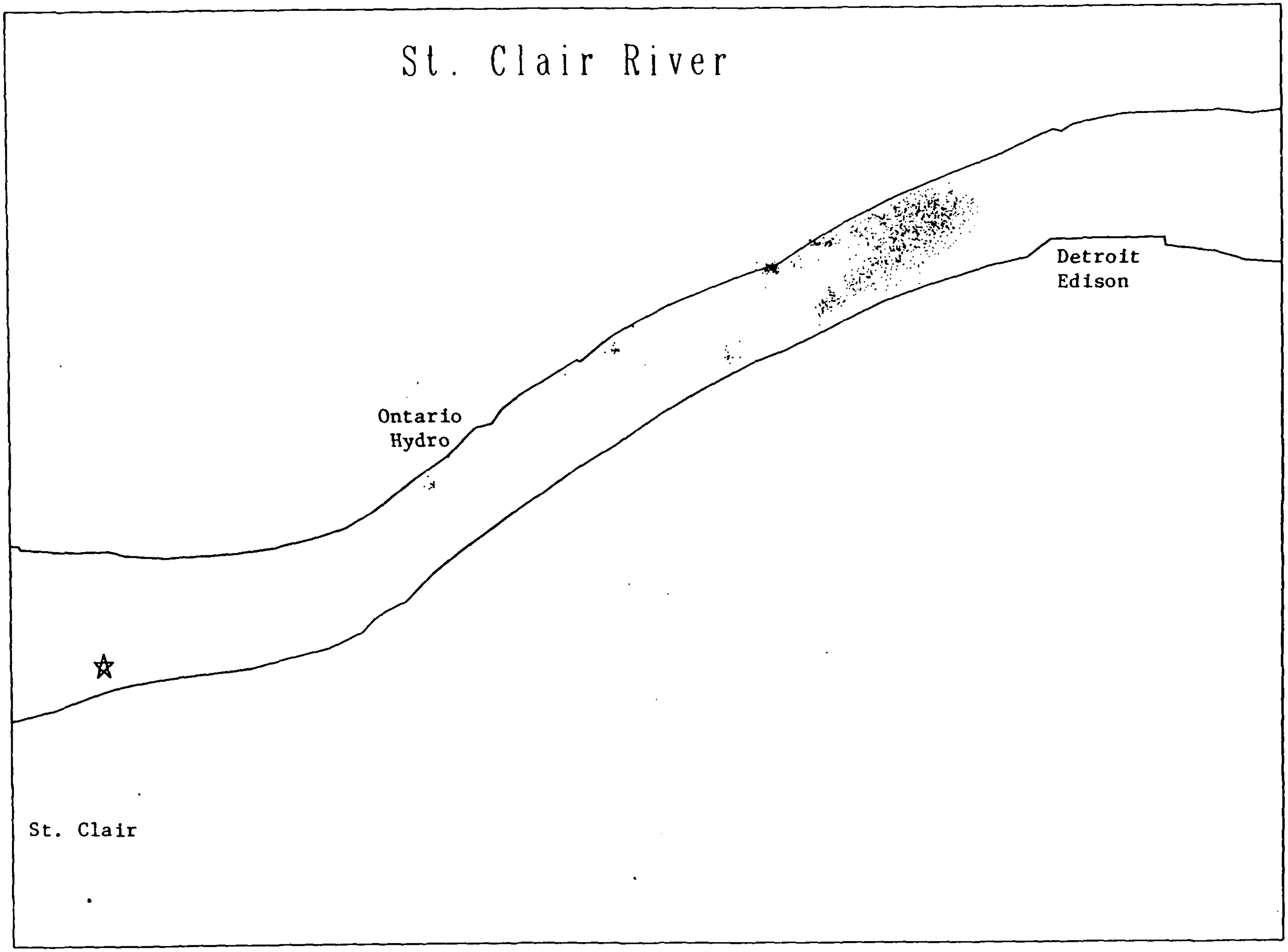


Fig. 14. Plot of slick location at t = 1 hr 45 min

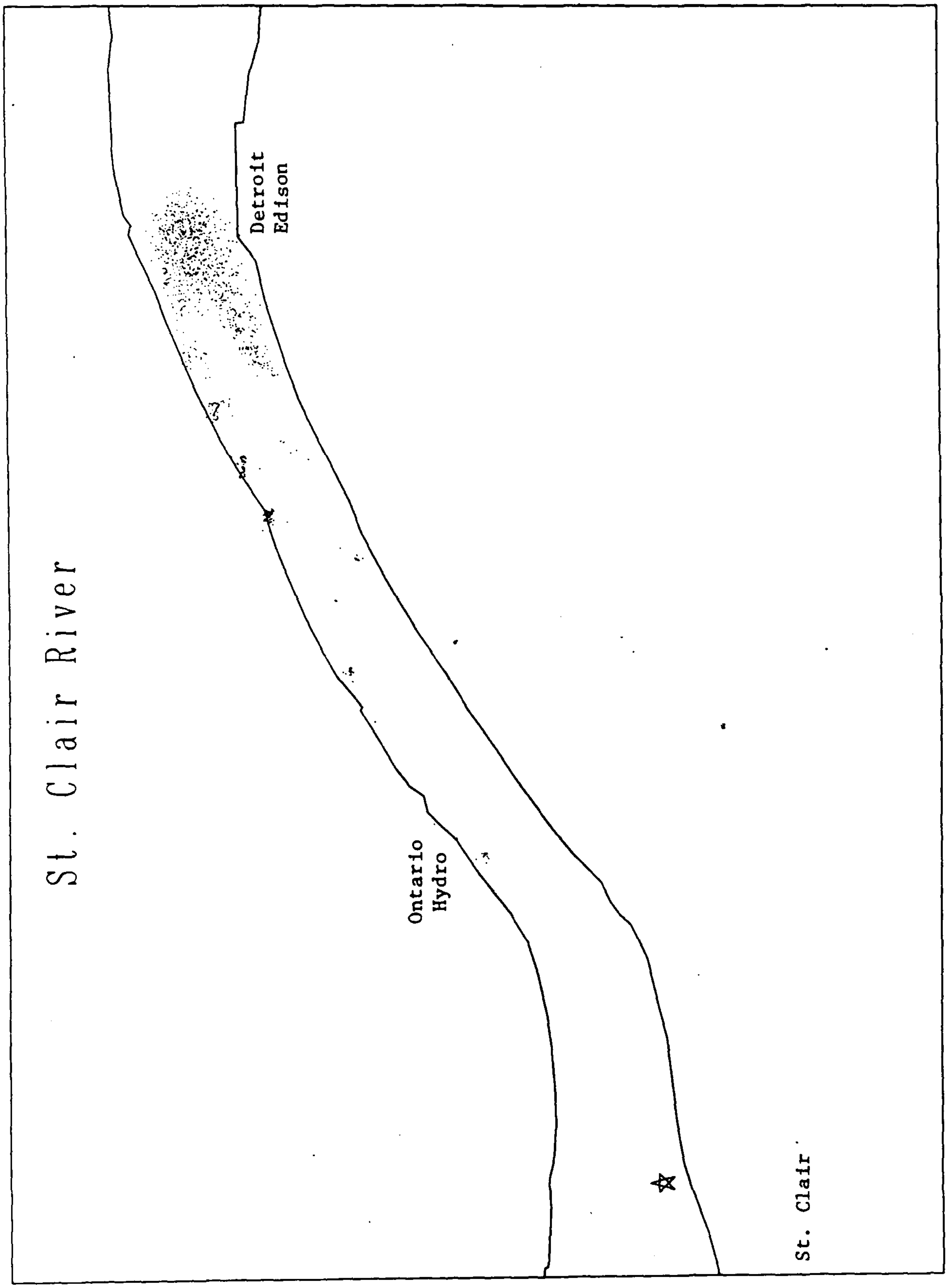


Fig. 15. Plot of slick location at t = 2 hrs

APPENDIX I

CROSS SECTIONAL GEOMETRY OF RIVERS

This appendix contains cross-sectional geometry data at each cross section in the following rivers:

1. St. Clair River
2. Detroit River
3. Upper St. Mary's River
4. Lower St. Mary's River

St. Clair River

X-Sect. No.		Vertical Line No. 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Distance(ft)	212	280	354	906	1230	1377	1625	1929											
	Depth (ft)	42	42	34	34	78	23	8	0											
2	Distance(ft)	185	343	451	600	806	839	998	1041	1115	1245	1401								
	Depth (ft)	11	38	32	32	39	52	69	60	42	7	0								
3	Distance(ft)	70	241	345	568	936	1020													
	Depth (ft)	24	42	38	67	6	0													
4	Distance(ft)	73	141	157	284	350	420	525	617	845	996									
	Depth (ft)	7	18	38	65	66	61	61	47	7	0									
5	Distance(ft)	18	76	116	266	362	465	529	619	710	809									
	Depth (ft)	10	18	33	45	46	57	52	42	24	0									
6	Distance(ft)	12	49	144	186	285	402	470	677	741	817	980								
	Depth (ft)	28	36	40	38	44	40	42	42	38	42	0								
7	Distance(ft)	11	56	141	223	310	367	502	575	820	873	909	977	1169	1227	1292				
	Depth (ft)	19	33	43	39	43	42	34	34	55	55	51	53	27	24	0				
8	Distance(ft)	54	135	179	462	590	895	1039	1178	1413	1458									
	Depth (ft)	24	31	48	41	35	52	44	30	33	0									
9	Distance(ft)	127	394	561	653	798	956	1062	1159	1276	1443									
	Depth (ft)	45	53	50	53	54	44	25	26	8	0									
10	Distance(ft)	10	155	272	444	537	855	995	1079	1336										
	Depth (ft)	14	46	43	50	46	46	33	32	0										
11	Distance(ft)	84	219	299	524	720	930	1147	1535	1613	1772	1782								
	Depth (ft)	35	40	38	49	44	46	28	27	19	9	0								
12	Distance(ft)	183	397	611	731	1312	1567	1719												
	Depth (ft)	36	33	41	42	33	23	0												
13	Distance(ft)	25	146	382	727	1009	1441	1749	1756											
	Depth (ft)	20	32	37	41	34	30	2	0											
14	Distance(ft)	127	639	1203	1432	1730	1956	2383	2486	2583										
	Depth (ft)	23	30	36	31	37	30	33	30	0										

Distance is measured from Lower Left Bank.

15	Distance(ft)	97	280	420	660	778	1119	1672	1842	2031	2275	2368	2549	2602
	Depth (ft)	27	32	29	36	31	28	34	36	32	46	19	15	0
16	Distance(ft)	180	384	582	720	840	1662	1794	2225	2419	2531	2618		
	Depth (ft)	25	33	29	35	29	30	27	40	24	19	0		
17	Distance(ft)	8	90	299	477	619	982	1556	1807	2123	2375	2466		
	Depth (ft)	28	31	28	40	29	37	33	38	38	18	0		
18	Distance(ft)	240	430	489	532	883	1134	1345	1763	1922	2055	2062		
	Depth (ft)	30	32	38	32	29	32	40	36	25	22	0		
19	Distance(ft)	103	122	241	711	1141	1443	1580	1724	1776	1853	1863		
	Depth (ft)	9	20	31	32	28	32	27	29	24	27	0		
20	Distance(ft)	54	292	682	1061	1201	1396	1773						
	Depth (ft)	13	35	31	39	37	47	0						
21	Distance(ft)	237	731	930	1176	1692	1806							
	Depth (ft)	33	38	35	41	24	0							
22	Distance(ft)	11	174	1203	1581	1757	1848	1943						
	Depth (ft)	10	32	37	33	20	4	0						
23	Distance(ft)	125	307	673	997	1644	1930	2028						
	Depth (ft)	25	26	35	33	36	20	0						
24	Distance(ft)	0	213	546	1186	1461	1683	1948	2103					
	Depth (ft)	7	26	29	41	34	36	27	0					
25	Distance(ft)	11	102	734	1184	1720	1910	1980						
	Depth (ft)	12	23	39	43	30	5	0						
26	Distance(ft)	60	140	734	835	1048	1864	2023						
	Depth (ft)	13	31	38	37	39	31	0						
27	Distance(ft)	119	577	1173	1737	2045	2192							
	Depth (ft)	22	35	34	33	24	0							
28	Distance(ft)	9	329	1083	1697	2085								
	Depth (ft)	10	32	36	32	0								
29	Distance(ft)	7	257	691	933	1174	1339	1671	1910					
	Depth (ft)	22	32	32	45	37	39	34	0					
30	Distance(ft)	168	1051	1296	1408	1605	1825	1953						
	Depth (ft)	29	40	38	35	36	22	0						

31	Distance(ft)	221	971	1294	1523	1675	2037										
	Depth (ft)	[illegible]	41	42	32	37	0										
32	Distance(ft)	6	138	320	1033	1877	2135	2166	2257								
	Depth (ft)	4	7	33	34	39	33	9	0								
33	Distance(ft)	39	470	836	912	1114	1753	1870	2026	2221	2350	2493	2680	2808	2924	3221	3507
	Depth (ft)	42	32	37	33	32	18	12	9	12	18	29	33	31	42	36	0
34	Distance(ft)	92	196	564	829	1068	2048	2132	2547								
	Depth (ft)	7	27	32	28	30	10	4	0								
35	Distance(ft)	5	218	314	405	700	920	1142	1236	1508	1601						
	Depth (ft)	6	32	29	31	28	31	26	26	14	0						
36	Distance(ft)	97	704	927	1033	1183	1537	1644									
	Depth (ft)	27	33	27	28	23	19	0									
37	Distance(ft)	229	657	1034	1174	1608	1643	1734	1903								
	Depth (ft)	25	31	29	24	19	23	4	0								
38	Distance(ft)	258	619	865	910	1053	1169										
	Depth (ft)	24	27	26	29	7	0										
39	Distance(ft)	361	507	758	953	1060	1169										
	Depth (ft)	26	25	31	28	8	0										
40	Distance(ft)	94	197	319	400	661	898	1069	1138	1203							
	Depth (ft)	4	24	22	24	28	30	18	2	0							
41	Distance(ft)	178	342	560	761	993	1107	1205									
	Depth (ft)	5	22	32	29	22	2	0									
42	Distance(ft)	41	225	1390	1429	2086	2395	2464	2872	3125	3512	3573	3608				
	Depth (ft)	4	34	30	23	4	3	3	38	30	22	6	0				
43	Distance(ft)	35	125	1390	1416	1776	1866	1977	2350	2624	2745	2963	3082				
	Depth (ft)	6	33	33	24	23	23	16	44	37	25	6	0				
44	Distance(ft)	78	264	1470	1729	1973	2206	2340	2505	2613							
	Depth (ft)	4	35	33	40	39	22	25	17	0							
45	Distance(ft)	101	212	389	616	864	1208	1389	1840	1994	2207	2385					
	Depth (ft)	4	31	36	31	38	31	41	36	25	17	0					
46	Distance(ft)	206	583	832	1046	1184	1743	2075									
	Depth (ft)	35	32	38	32	39	32	0									

47	Distance(ft)	9	108	201	656	935	1696	1924	2002	2139					
	Depth (ft)	3	6	36	31	39	39	20	5	0					
48	Distance(ft)	130	265	941	1318	1479	2018	2209	2352						
	Depth (ft)	3	34	40	28	39	40	5	0						
49	Distance(ft)	67	174	879	1101	1316	1815	2064	2263	2353	2555	2561			
	Depth (ft)	9	33	36	27	26	38	40	28	3	3	0			
50	Distance(ft)	43	166	272	1126	1228	1336	1866	2008	2706	2078	2831	3061	3078	
	Depth (ft)	5	6	32	34	18	26	26	34	30	79	6	4	0	
51	Distance(ft)	20	110	256	1159	1183	1311	1453	1889	2836	3127	3222	3393		
	Depth (ft)	4	5	33	36	20	11	19	28	33	27	7	0		
52	Distance(ft)	124	208	666	1089	1183	1579	1722	2014	2398	2791	3111	3357	3498	
	Depth (ft)	7	31	32	32	10	9	23	28	19	35	36	8	0	
53	Distance(ft)	31	439	726	988	1097	1501	1673	2106	3066	3309	3419			
	Depth (ft)	23	41	32	31	6	7	35	21	36	7	0			
54	Distance(ft)	113	992	1152	1300	1560	1650	2257	2424	3050	3218	3339			
	Depth (ft)	33	32	6	40	34	23	22	29	30	6	0			
55	Distance(ft)	210	661	1131	1311	1757	2221	2771	2922						
	Depth (ft)	34	34	46	24	17	30	30	0						
56	Distance(ft)	47	232	400	520	703	781	1081	1175	1287	1467	1662	2219	2259	2553
	Depth (ft)	23	32	41	35	34	43	16	18	14	23	24	28	27	0
57	Distance(ft)	95	330	480	880	1216	1259	1593	2091	2302					
	Depth (ft)	8	49	49	34	32	27	33	30	0					
58	Distance(ft)	57	263	480	902	1838	2185								
	Depth (ft)	25	43	46	34	30	0								
59	Distance(ft)	77	296	895	997	1117	1331	1736	1911	2068	2087				
	Depth (ft)	5	36	53	44	46	35	31	7	5	0				
60	Distance(ft)	86	498	984	1181	1551	1637	1701	1704						
	Depth (ft)	27	40	38	50	27	6	3	0						
61	Distance(ft)	1	73	308	436	612	793	1017	1387	1548	1700				
	Depth (ft)	3	5	50	44	50	45	53	37	6	0				
62	Distance(ft)	338	700	880	1436	1806									
	Depth (ft)	45	45	53	25	0									

63	Distance(ft)	375	906	1171	1452	1580	1726									
	Depth (ft)	41	60	55	28	26	0									
64	Distance(ft)	250	595	1047	1270	1556	1675									
	Depth (ft)	47	52	46	28	8	0									
65	Distance(ft)	178	524	828	1281	1450	1594									
	Depth (ft)	21	44	57	49	7	0									
66	Distance(ft)	117	739	1278	1448	1532	1633	1667								
	Depth (ft)	26	53	45	21	6	4	0								
67	Distance(ft)	140	354	901	1172	1611	1765	1935								
	Depth (ft)	20	33	47	50	35	11	0								
68	Distance(ft)	0	39	953	1358	1578	1666	1863	1910							
	Depth (ft)	20	27	48	47	23	5	4	0							
69	Distance(ft)	10	608	951	1765	2026	2351	2376								
	Depth (ft)	27	42	43	30	6	4	0								
70	Distance(ft)	125	298	745	973	1286	1660	2050	2230	2331						
	Depth (ft)	5	32	42	40	47	34	30	7	0						
71	Distance(ft)	146	247	628	1090	1433	1750	2094	2339	2718	2727					
	Depth (ft)	5	28	36	38	49	33	32	6	4	0					
72	Distance(ft)	20	196	297	1061	1604	2228	2476	2535	2820						
	Depth (ft)	4	5	31	33	40	32	24	6	0						
73	Distance(ft)	21	183	324	1295	1878	2193	2665	2714	2969	2997					
	Depth (ft)	5	5	27	37	34	36	23	6	4	0					
74	Distance(ft)	10	71	254	838	1430	1907	2387	2560	2721	2904	2912				
	Depth (ft)	3	3	27	31	34	47	27	28	6	3	0				
75	Distance(ft)	88	319	921	1143	1418	1731	2458	2602	3101						
	Depth (ft)	21	32	35	46	43	29	24	5	0						
76	Distance(ft)	79	492	820	1034	1389	2126	2424	2531	2809	3067	3291	3293			
	Depth (ft)	27	46	44	33	26	25	26	8	5	7	4	0			
77	Distance(ft)	153	289	472	1112	1202	1820	2030	2194	2406	2550	2677	2858	3071	3241	3307
	Depth (ft)	42	45	38	33	24	19	24	37	28	27	8	5	8	7	0
78	Distance(ft)	117	660	1208	1403	1579	1753	2065	2494	3000	3460	3740	3935			
	Depth (ft)	27	33	27	6	6	18	18	2	6	20	4	0			

79	Distance(ft)	312	545	1519	1792	2299	3429	3584	4013	4167	4286			
	Depth (ft)	6	27	27	29	0	0	27	26	2	0			
80	Distance(ft)	236	388	539	1174	1288	1974	2128	2206	2394	2631	2722	2855	2864
	Depth (ft)	38	33	38	35	28	32	8	27	45	31	6	5	0
81	Distance(ft)	133	669	1107	1234	1560	1873	2322	2512	2574	2823			
	Depth (ft)	32	37	36	24	23	40	53	26	4	0			
82	Distance(ft)	6	220	666	1235	1304	1438	1700	1940	2256	2519	2744		
	Depth (ft)	5	31	37	34	28	41	27	48	43	6	0		
83	Distance(ft)	66	340	1082	1349	1712	1954	2201	2302	2426				
	Depth (ft)	5	34	35	48	48	30	23	6	0				
84	Distance(ft)	64	169	404	910	1013	1428	1941	2212	2350				
	Depth (ft)	6	6	27	42	52	53	28	8	0				
85	Distance(ft)	69	136	359	596	1138	1526	2042						
	Depth (ft)	5	18	31	33	59	56	0						
86	Distance(ft)	7	161	329	717	992	1318	1534	1810	1928	2071			
	Depth (ft)	3	5	33	35	53	46	56	41	8	0			
87	Distance(ft)	2	49	110	309	587	775	978	1289	1531	1782	1879	2008	
	Depth (ft)	5	6	18	34	35	40	53	44	48	26	5	0	
88	Distance(ft)	97	193	397	686	883	1092	1369	1782	2072	2181			
	Depth (ft)	5	22	34	33	38	51	47	49	7	0			
89	Distance(ft)	215	446	852	1345	1764	2098	2296						
	Depth (ft)	24	33	36	53	51	5	0						
90	Distance(ft)	98	225	459	822	1598	1941	2270						
	Depth (ft)	5	42	62	44	54	7	0						
91	Distance(ft)	50	239	417	1551	2049	2137	2615						
	Depth (ft)	5	51	54	31	28	6	0						
92	Distance(ft)	175	660	1554	2044	2180	2583							
	Depth (ft)	46	47	25	22	6	0							
93	Distance(ft)	83	217	386	702	876	1051	1348	1707	1919	2108	2246		
	Depth (ft)	3	32	38	45	57	49	46	26	27	5	0		
94	Distance(ft)	255	787	1272	1574	1758	1988	2090	2372					
	Depth (ft)	33	34	47	44	36	33	7	0					

95	Distance(ft)	48	403	884	1581	1702	1990	2250	2671								
	Depth (ft)	10	33	44	41	35	38	6	0								
96	Distance(ft)	1	105	268	594	1211	1459	1697	2152	2394	2491	2932					
	Depth (ft)	5	4	41	39	33	37	32	46	29	5	0					
97	Distance(ft)	31	131	569	914	1125	1528	2138	2356	2520	2547	2831	3240	3556	3681	3963	3973
	Depth (ft)	25	40	46	34	31	11	5	6	16	33	42	36	36	15	11	0

Detroit River

X-Sect. No.		2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Distance(ft)	25	125	500	1000	1200	1750	2000	2250	3000	3425									
	Depth (ft)	14	28	36	37	33	32	30	10	9	0									
2	Distance(ft)	155	900	1025	1275	1400	1700	2125	2375	3250	3375									
	Depth (ft)	25	28	36	40	36	43	30	10	5	0									
3	Distance(ft)	50	125	750	1250	1350	2425	2525	3250	3425										
	Depth (ft)	12	20	21	22	28	28	5	4	0										
4	Distance(ft)	75	175	250	825	1400														
	Depth (ft)	12	22	27	27	0														
5	Distance(ft)	75	175	250	825	1250														
	Depth (ft)	12	22	27	27	0														
6	Distance(ft)	81	142	198	373	627	760	1124	1184	1233	1322									
	Depth (ft)	11	26	29	30	31	30	24	21	10	0									
7	Distance(ft)	125	250	500	800	1000	1250	2050	2850	3500	4000	4250	4650	4825	5000	5100				
	Depth (ft)	12	25	21	20	28	23	23	38	38	29	43	45	7	7	0				
8	Distance(ft)	75	275	575	1250	2075	2825	3825	4450	4600	4675	5075								
	Depth (ft)	4	18	24	18	12	27	29	39	18	2	0								
9	Distance(ft)	101	266	507	756	1223	1360	1461												
	Depth (ft)	25	25	29	33	31	26	0												
10	Distance(ft)	25	75	475	575	800	1250	1875	2050	2550	2750	3400								
	Depth (ft)	12	25	23	12	6	2	5	22	27	5	0								
11	Distance(ft)	88	159	1107	1093	1998	2074	2111												
	Depth (ft)	22	27	21	26	23	12	0												
12	Distance(ft)	2	135	550	695	883	1000	1374	1692	1844	1923									
	Depth (ft)	10	25	31	24	29	20	22	26	21	0									
13	Distance(ft)	24	607	1005	1208	1388	1598	1684	1662	1891	2194	2204								
	Depth (ft)	20	22	29	10	33	27	20	13	7	6	0								
14	Distance(ft)	125	250	350	680	825	1250	1525	1625	1825	2125	2533	2550							
	Depth (ft)	26	35	33	48	42	46	35	35	9	8	9	0							

Distance is measured from Lower Left Bank.

15	Distance(ft)	109	317	529	755	1004	1660	1831	1921							
	Depth (ft)	31	39	35	40	37	42	24	0							
16	Distance(ft)	30	63	250	475	563	1125	1375	1625	1750	1825	1950	2000			
	Depth (ft)	5	14	15	42	32	33	41	43	35	38	23	0			
17	Distance(ft)	125	375	425	1000	1250	1375	1500	1725	1850	2000	2375	2750	3125	3500	3750
	Depth (ft)	12	18	25	35	37	37	9	9	25	37	29	28	32	35	0
18	Distance(ft)	10	550	625	1375	1750	2125	2310	2625	2910						
	Depth (ft)	27	28	37	36	36	48	43	38	0						
19	Distance(ft)	118	369	634	811	1338	1657	2224	2373							
	Depth (ft)	26	22	32	51	39	48	41	0							
20	Distance(ft)	25	125	200	475	600	750	1000	1250	1600	1750	2000	2125	2250		
	Depth (ft)	23	28	24	34	30	45	50	43	47	36	45	42	0		
21	Distance(ft)	125	175	375	1375	1500	1625	1800	1875	2050						
	Depth (ft)	18	29	41	45	47	32	18	6	0						
22	Distance(ft)	197	451	717	1187	1594	1674	1903								
	Depth (ft)	38	46	47	40	43	39	0								
23	Distance(ft)	25	125	1000	1500	2250	2500	2550								
	Depth (ft)	28	38	38	31	32	27	0								
24	Distance(ft)	25	510	1120	2075	2375	2600	2710	2825							
	Depth (ft)	9	45	38	35	38	34	8	0							
25	Distance(ft)	25	125	300	1000	1500	1750	2375	2625	2650						
	Depth (ft)	19	25	40	39	32	36	36	24	0						
26	Distance(ft)	250	680	1125	1485	1750	2090	2500	2850	3320	3750	4250				
	Depth (ft)	38	43	36	35	41	37	38	10	5	8	0				
27	Distance(ft)	175	1050	2175	2500	2675	3300	3425	4425	4675						
	Depth (ft)	34	38	28	18	25	27	4	3	0						
28	Distance(ft)	185	375	500	625	850	1150	1500	1650	2000	2125	2310	2450	2510		
	Depth (ft)	29	40	41	40	36	37	36	38	30	9	5	7	0		
29	Distance(ft)	27	231	450	836	1548	1842	2331	2587	2865	3354	3400				
	Depth (ft)	3	6	20	30	34	46	31	34	8	7	0				
30	Distance(ft)	117	223	804	997	1503	1829	1976	2551	3197	3392	4179	4208	4277	4777	
	Depth (ft)	35	37	30	11	6	10	31	39	35	17	5	9	8	0	

31	Distance(ft)	33	115	327	593	864	1149	1199	1254	1300	1337	2001	2390	2443	2984	3235	4076	4409	6204	6715
	Depth (ft)	18	35	34	36	9	6	10	0	0	8	33	7	6	14	38	38	9	6	6
32	Distance(ft)	300	500	780	1310	1790	2090	2900	3250	3680	5600	5650	5750							
	Depth (ft)	31	31	5	5	35	27	35	36	7	7	9	0							
33	Distance(ft)	450	575	1250	1750	2700	2725	2825	4500	5000	5025									
	Depth (ft)	6	28	36	33	28	18	5	3	2	0									
34	Distance(ft)	120	174	461	851	986	1040	1106	1175											
	Depth (ft)	9	20	35	32	22	13	5	0											
35	Distance(ft)	24	573	722	896	976														
	Depth (ft)	14	33	31	16	0														
36	Distance(ft)	199	502	854	952	1887	1980	2265	2428											
	Depth (ft)	24	26	24	9	7	26	21	0											
37	Distance(ft)	92	301	446	681	969	1052	1707	2243	2375										
	Depth (ft)	25	31	28	35	22	8	7	4	0										
38	Distance(ft)	1500	1625	2000	2100	2300	3750	3825	3900	4225	4375	4675								
	Depth (ft)	0	25	25	2	0	0	5	32	32	2	0								
39	Distance(ft)	25	625	750	1125	1175	1250	1750	1925	2125	2150									
	Depth (ft)	2	1	27	27	22	30	32	6	2	0									
40	Distance(ft)	50	475	1000	2000	2100	2800	3100	3300	3750	4250	4500	5100	5500	5900	6900	7000	7100		
	Depth (ft)	3	3	27	27	35	35	25	32	30	33	28	28	9	8	8	12	0		
41	Distance(ft)	73	293	2564	2710	3746	3772	4424	450[illegible]	4996	5151	5328	5482	5796	6111	6346				
	Depth (ft)	3	6	11	27	24	29	36	27	25	22	28	28	11	13	0				
42	Distance(ft)	575	750	1000	2750	3175	3375	4300	4375	4950	5400	5900	6050							
	Depth (ft)	6	6	5	3	6	18	18	27	27	18	6	0							
43	Distance(ft)	17	129	284	474	657	812	867	898											
	Depth (ft)	5	31	36	37	30	9	7	0											
44	Distance(ft)	4	45	244	536	779	993	1152	1850	1940	2003									
	Depth (ft)	3	4	32	33	31	7	4	3	2	0									
45	Distance(ft)	55	278	434	460	502	773	823	848	97[illegible]	1095	1246								
	Depth (ft)	19	18	23	15	33	35	21	25	11	8	0								
46	Distance(ft)	16	70	240	440	591	876	1238	1377	1496	1523									
	Depth (ft)	2	3	22	10	10	27	24	8	[illegible]	0									

47	Distance(ft)	48	100	636	756	838	982	1144	1165	1379	1425								
	Depth (ft)	4	24	16	20	18	30	27	20	15	0								
48	Distance(ft)	31	209	990	1146														
	Depth (ft)	25	29	27	0														
49	Distance(ft)	226	275	766	1246	1345	1806	1922	2396	2472	2639								
	Depth (ft)	8	7	8	22	19	16	20	16	10	0								
50	Distance(ft)	320	406	559	882	1033	1508	1676	2128	2194	2468	2542	2646	2709	2771	3015	3209	3432	3675
	Depth (ft)	20	8	9	23	8	5	22	15	9	10	17	14	8	7	2	6	7	0
51	Distance(ft)	875	1000	2000	2175	2550	4125	4625	4875	5125	5500	6500							
	Depth (ft)	4	18	18	5	0	0	2	14	14	4	0							

Upper St. Mary's River

X-Sect. No.		2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Distance(ft)	4	400	800	2000	3250	4050	7500	8950	9480	11400	12080	12900	14300	18800	21790	21800			
	Depth (ft)	2	6	12	18	24	30	28	28	40	30	24	18	12	6	4	0			
2	Distance(ft)	10	1030	1350	5000	6600	8000	10000	10230	12100	14000	15400	18000	19750	19900	19990	20000	22950		
	Depth (ft)	5	6	12	18	20	28	28	30	24	18	6	2	0	0	1	1	0		
3	Distance(ft)	10	550	1220	1320	1420	1700	2000	3000	7000	9000	11000	11200	12120	13550	16050	22340	22350		
	Depth (ft)	5	6	12	18	24	30	46	77	29	28	28	46	30	24	12	3	0		
4	Distance(ft)	10	250	500	650	800	1100	7350	9200	10950	11180	11280	14280	18000	19590	19600				
	Depth (ft)	5	6	12	18	24	27	55	46	28	24	18	12	6	5	0				
5	Distance(ft)	4	580	880	3220	5500	6600	8100	10200	11000	13900	14080	14180	14600	17300	18050	20520	22400		
	Depth (ft)	2	6	12	18	21	18	18	24	28	28	30	24	12	14	12	6	0		
6	Distance(ft)	12	830	1800	3920	5190	5350	6750	7000	7980	8150	8180	8400	12300	14390	14400				
	Depth (ft)	6	6	13	16	24	28	28	38	30	24	18	12	6	6	0				
7	Distance(ft)	8	1150	2000	2150	3600	3700	4100	4600	6500	7970	7980								
	Depth (ft)	4	6	12	28	28	17	12	8	6	4	0								
8	Distance(ft)	6	420	2600	5700	6240	6250	6900	6910	8150	8200	8250	9500	9600	9650	12090	12100			
	Depth (ft)	3	6	10	6	3	0	0	4	6	12	28	23	12	6	4	0			
9	Distance(ft)	6	1880	1950	2650	2680	3400	3500	4700	4800	4890	4900								
	Depth (ft)	3	3	0	0	6	12	28	28	12	6	0								
10	Distance(ft)	4	880	1000	1050	2300	2400	2450	2550	2570	2580									
	Depth (ft)	2	6	12	25	28	24	18	6	3	0									
11	Distance(ft)	4	680	820	950	1050	1250	2450	2480	2680	2880	3800	5200	8000	8400	9770	9780			
	Depth (ft)	2	6	18	24	30	26	28	24	18	12	8	10	10	6	5	0			
12	Distance(ft)	4	760	850	960	1150	1600	2650	3350	3650	3900	4000	4950	5700	6200	7396	7100			
	Depth (ft)	2	6	18	24	30	28	28	30	24	18	12	9	13	6	2	0			
13	Distance(ft)	4	450	550	680	780	1200	1650	3100	3350	3680	3950	5900	6500	7200	7380	8940	8950		
	Depth (ft)	2	6	12	18	24	30	28	28	30	24	18	13	5	9	6	4	0		
14	Distance(ft)	8	730	850	1180	1380	1680	2180	3700	3900	4250	4530	4550							
	Depth (ft)	4	6	12	18	24	30	28	28	24	18	12	0							

Distance is measured from Lower Left Bank.

15	Distance(ft)	4	1000	1950	2050	3450	3900	4180	4400	5300	6180	6350	6560	6580				
	Depth (ft)	2	5	24	28	28	6	30	34	39	30	24	12	0				
16	Distance(ft)	5	350	355	430	435	750	755	1100	1180	2300	3200	3900	4100	4400	5200	5750	
	Depth (ft)	28	28	0	0	28	28	0	0	14	12	6	6	24	12	6	0	
17	Distance(ft)	40	185	300	450	550	650	750	1000	1250	1300	1400	2400	2450	4350	4400	4700	4750
	Depth (ft)	28	0	28	0	28	0	28	0	3	0	3	3	0	0	3	3	0

Lower St. Mary's River

X-Sect. No.		Vertical Line No. 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Distance(ft)	133	267	2067	2267	2400	2667	2800												
	Depth (ft)	12	28	28	16	12	6	0												
2	Distance(ft)	333	467	667	800	2000	2133	2733	3067	[illegible]										
	Depth (ft)	6	8	6	28	28	19	12	6	6										
3	Distance(ft)	10	250	375	1200	1350	1950	2125	2310	3060	3250									
	Depth (ft)	5	11	35	37	24	23	29	18	8	0									
4	Distance(ft)	11	440	563	733	804	1025	1093	1195	1297	1372	1614	1781	2000						
	Depth (ft)	6	5	6	17	5	12	33	35	27	32	4	3	0						
5	Distance(ft)	1024	1156	1209	1396	1460	1515	1613	1651	1764	1803	1879	1970							
	Depth (ft)	3	9	7	28	24	34	27	30	25	29	2	0							
6	Distance(ft)	15	192	437	622	820	1027	1120	1387	1493	1863	2057	2399	2416						
	Depth (ft)	1	3	25	13	27	15	26	18	29	11	3	3	0						
7	Distance(ft)	15	399	657	1103	1334	1360	1517	1554											
	Depth (ft)	3	12	38	54	4	8	5	0											
8	Distance(ft)	259	400	504	618	688	809	918	1168	1327	1687	1919	2027	2458	2890					
	Depth (ft)	6	3	12	6	12	3	4	34	25	30	21	24	16	0					
9	Distance(ft)	267	533	800	1000	1067	1533	1600	1667	2000	2200	2800								
	Depth (ft)	6	12	12	18	28	28	18	12	12	6	0								
10	Distance(ft)	142	173	336	591	703	738	861	977	1070	1180									
	Depth (ft)	5	17	17	34	34	30	47	39	3	0									
11	Distance(ft)	267	333	400	433	733	1000	2000	2667	3800	4200	4333	5867							
	Depth (ft)	6	12	18	26	26	3	1	3	20	12	7	0							
12	Distance(ft)	467	733	800	1867	2000	2667	5733												
	Depth (ft)	6	12	13	13	6	2	0												
13	Distance(ft)	31	297	829	1006	1272	1448	1803	2069	2246	2335	2366	3192							
	Depth (ft)	4	5	26	16	18	8	7	15	15	13	12	0							
14	Distance(ft)	467	600	667	867	1867	2667	2867												
	Depth (ft)	6	18	19	18	12	6	0												

Distance is measured from Lower Left Bank.

15	Distance(ft)	58	105	346	370	410	517	595	615									
	Depth (ft)	5	16	37	36	38	32	2	0									
16	Distance(ft)	67	200	333	467	600	633	867	900	1033								
	Depth (ft)	6	12	17	12	18	30	30	18	0								
17	Distance(ft)	70	490	712	770	912	1067	1133	4333	4400								
	Depth (ft)	31	35	32	5	0	0	1	1	0								
18	Distance(ft)	333	467	533	667	2067	2117	2177	2287	2442	2592	2712	2827	3333	3600	3800		
	Depth (ft)	3	10	3	0	0	8	30	30	33	30	30	6	12	6	0		
19	Distance(ft)	5	68	175	250	374	400	447	867	1200	1333	1400	1933	2000	2133	2667		
	Depth (ft)	2	3	21	22	14	5	0	0	6	12	27	27	12	6	0		
20	Distance(ft)	94	155	227	405	530	915	950	1188	1368	1500							
	Depth (ft)	8	7	32	37	33	34	15	5	4	0							
21	Distance(ft)	500	625	1000	1300	1450	1550	1625	2625	3375	3400							
	Depth (ft)	5	30	34	31	13	16	4	5	3	0							
22	Distance(ft)	569	791	1461	1690	2342	2662	3030	4419	4546	4758	4865	5117	5273	5475	5760		
	Depth (ft)	10	30	32	7	13	0	8	10	6	14	6	30	29	8	0		
23	Distance(ft)	625	750	1000	1625	1750	3250	3350	3625	3750	4125	4300	4500	5250	5375	5675	6125	
	Depth (ft)	5	15	15	25	35	35	25	26	21	26	19	21	14	5	6	0	
24	Distance(ft)	1875	2125	2300	3050	3575	3690	4600	4690	5050	5125	5225	6060	6175				
	Depth (ft)	16	8	35	32	37	35	35	28	18	4	11	3	0				
25	Distance(ft)	300	750	1500	2250	2550	2600	3950	4125	4450	5250	5375	5800					
	Depth (ft)	2	10	5	20	20	33	35	40	35	35	5	0					
26	Distance(ft)	2800	3050	3250	3750	4000	4900	5200	7400	8250								
	Depth (ft)	13	40	17	19	9	37	20	8	0								
27	Distance(ft)	50	800	1250	1400	2000	2150	2875	4100	4750	4800	5200	5260	5900	6450	7000	9250	9500
	Depth (ft)	2	5	5	38	31	10	5	12	12	33	30	12	13	8	8	8	0
28	Distance(ft)	1933	2000	2333	2400	7667	8667	8700	9000	9033	9300	9333	10467	12000				
	Depth (ft)	1	27	27	2	6	12	27	27	21	21	12	6	0				
29	Distance(ft)	310	1875	1935	2125	2250	2325	2935	3000									
	Depth (ft)	4	6	32	35	31	6	4	0									
30	Distance(ft)	125	1150	1500	1575	1875	2000	2250	2560	2650								
	Depth (ft)	5	8	9	32	32	8	8	5	0								

31	Distance(ft)	400	1100	1700	2050	2200	2625									
	Depth (ft)	7	10	42	20	20	0									
32	Distance(ft)	60	210	335	350	410	425									
	Depth (ft)	32	35	33	36	19	0									
33	Distance(ft)	10	290	300												
	Depth (ft)	33	33	0												
34	Distance(ft)	6	18	190	290	302										
	Depth (ft)	19	32	31	32	0										
35	Distance(ft)	10	125	290	302											
	Depth (ft)	32	33	33	0											
36	Distance(ft)	40	80	250	385	425										
	Depth (ft)	10	35	35	35	0										
37	Distance(ft)	780	935	1060	1125	1435	1500	1625	1750							
	Depth (ft)	10	5	16	35	33	27	28	0							
38	Distance(ft)	15	125	200	300	600	675	800	875	935	1375	1625	1930	2000	2125	2375
	Depth (ft)	13	17	14	35	36	27	27	0	6	7	4	9	5	5	0
39	Distance(ft)	185	250	500	600	810	1000	2375	2525							
	Depth (ft)	7	32	32	27	25	6	5	0							
40	Distance(ft)	125	280	310	840	875	1250	1375	810							
	Depth (ft)	16	18	36	33	26	23	5	0							
41	Distance(ft)	90	125	375	600	925	1300	1690	1750	2080	2250	2560	2600			
	Depth (ft)	30	31	35	32	40	32	32	13	9	4	4	0			
42	Distance(ft)	1825	2110	2325	2700	2825	3100	3575	3725	4075	4200	4313				
	Depth (ft)	1	11	8	11	29	38	30	12	3	3	0				
43	Distance(ft)	500	2500	2625	2750	3125	3250	3750	4750	5000						
	Depth (ft)	6	14	33	36	32	20	8	5	0						
44	Distance(ft)	500	690	810	1225	1310	1600	1850	2050	2310	3025	3625	3750	3810		
	Depth (ft)	5	10	6	9	13	13	30	31	16	22	4	4	0		
45	Distance(ft)	190	510	625	825	1500	1650	2250	2600	2910	3125	3350				
	Depth (ft)	2	18	15	36	35	8	8	12	3	3	0				
46	Distance(ft)	125	250	415	610	725	775	950	1125	1225	1275	1350				
	Depth (ft)	10	35	32	36	34	18	20	18	10	12	0				
47	Distance(ft)	50	125	225	370	590	685	710	1000	1025	1375	1600	1875			
	Depth (ft)	12	16	12	26	28	21	36	35	27	24	6	0			

APPENDIX II

PROGRAM LISTING OF ROSS AND SUBPROGRAMS

The program listing is arranged in the following sequence.

Main Program

ROSS

ROSS Subroutines

ADVECT
BOUNDR
DISOLU
EVAPOR
NDCONV
ORIENT
PLOTNU
PRELSE
PRINT1
SPRDAX
SPRD1X
SPRD1Y
VELDIS

System Subroutines[1]

GAUSS
RANDU

[1]These programs were available at Clarkson University computing system. The source code is provided here for completeness. Other systems may substitute these with appropriate subprograms.

Main program ROSS

```
C
C     ************************************************************************
C     *
C     *            River Oil Slick Simulation Model              ..ROSS..
C     *
C     *      Last Date of Revision October 14, 1986
C     *
C     *    Developed by the Department of Civil and Environmental Engineering
C     *                 Clarkson University, Potsdam, New York 13676
C     *      under the support of the Detroit District, U. S. Army Corps of
C     *      Engineers, through the Cold Regions Research and Engineering
C     *      Laboratory, Hanover, N.H.
C     *----------------------------------------------------------------------------
C     *  This program is furnished by the Government and is accepted and used
C     *  by the recipient upon the express understanding that
C     *  the United States Government University makes no warranties, express
C     *  or implied, concerning the accuracy, completeness, reliability,
C     *  usability, or suitability for any particular purpose of the
C     *  information and data contained in this program or furnished in
C     *  connection therewith, and the United States Government shall be under
C     *  no liability whatsoever to any person by reason of any use made
C     *  thereof.  The program herein belongs to the Governmeent.  Therefore,
C     *  the recipient further agrees not to assert any proprietary rights
C     *  therein or to represent this program to anyone as other than a
C     *  Government program.
C     *----------------------------------------------------------------------------
C     *
C     ************************************************************************
C
C
      DIMENSION IPARTX(5),IPARTY(5),HLIFE(10),THETA0(4),IDUM(20)
      COMPLEX VSTRM(99,16),CORDV(99,16),VCAR(12000),CORDLB(99)
      COMPLEX SPCEN,PARTCL(1000),VWIND,VDRIFT
      COMPLEX SPCENO
      COMMON /VEL/VSTRM,CORDV,CORDLB,Q(30),WL(30),TICE(99,20),
     $ YWID(99,20),Z(99,20),ZD(99),NSLSCT(99),SCTANG(99),
     $ LCSTSQ(30),NSTUBE(99),NUMCON(99),NFIRCO(99),NSECO(99),KINTM
      COMMON /VA/ VCAR,VWIND,VDRIFT
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      COMMON /BLOCK7/SPGOIL,ANIU,SIGMA,AK2I,AK2V,AK2T,
     $ VOLPAR,VOLPIE(8),SLICKR(8)
      COMMON /BLOCK8/AKC10,AKC20,AKC30
      COMMON /SE/FEVP1,FEVP2,CEVP,T0EVP
      COMMON /V/IZRBX(100),IZRBY(100),NZRVB
      COMMON /ICE/NICEX1(20),NICEY1(20),NICEX2(20),NICEY2(20),NICERG,
     $ AMIUO,ANICE,IPOS1(20),IPOS2(20),SPAICE
      COMMON /SO/IMOVIN(1000),YSHIF1(1000),NMOVIN,SSHIFT
      INTEGER UFSTPS,OSTPS
      character *4 FULL,PART,OPEN,STCL,DETR,STMU,STML,WORD
      character *12 finame
      character *4 TEXT(11),FUELTP(4)
      character *44 SLINFO(3)
      EXTERNAL RANDOMIZE
      DATA FULL,PART,OPEN/'FULL','PART','OPEN'/
```

```
      DATA STCL,DETR,STMU,STML/'STCL','DETR','STMU','STML'/
      DATA HLIFE/0.033,.5,1.,6.,12.,18.,24.,48.,48.,8760./
      DATA THETA0/109.,127.1,6.0,27.2/
      DATA SLINFO(1)/' Pie  No. of particles  Mean radius(ft)'/
      DATA SLINFO(2)/' Strip  Particles  -Le(ft)   Y mean   Le(ft)'/
      DATA SLINFO(3)/' Strip  Particles  -Le(ft)   X mean   Le(ft)'/
        open(15,file='ross.fnm')
        rewind 15
        do 2222 ifiles=1,9
        read(15,1111)iunit,finame
        open(iunit,file=finame)
        rewind iunit
2222    continue
1111    format(I3,A12)
C   Explanation of Variables
C   --single varibles
C
C     The next four variables are for controlling output.  They can
C     have the values  0-NO , 1-YES
C     IOPT1-Fixed data like Geometry and Bank (shore) conditions
C     IOPT2-Computed Vleocities for plotting
C     IOPT3-Location of particles for plotting
C     IOPT4-Number Plot(particle distribution) on print
C
C     ISPTYP - Spill type  0-Instantaneous,  1-Continuous.  Computed by
C              model based on : if SPLTIM > 0.5*SPILDT  ISPTYP=1 else=0
C
C     FEVP1  - Fraction evaporated at previous time step
C     FEVP2  - Fraction evaporated at present time step
C
C     NBRNCH  No. of Branches in the 1-D Flow Model
C     NGRIDX  Total No. of grid boxes in X-direction
C
C     TOTDIS  - Total amouunt of Dissolved Oil (gms)
C
C     UFDT    -  1-D Model time step(hrs)
C     UFSTPS  - No. of 1-D model steps
C     OSTPS   - No. of Oilspill steps per UFDT
C
C     ** Any variable with UNI as the last three characters are for
C     converting units for the purpose of printing
C
C   --some important variable names used for intermediate computations
C     AIY  area of the IYth Trapezoid
C     PERI wetted perimeter by IYth Trapezoid
C     HR   hydraulic radius of IYth trapezoid
C
C   --One D arrays
C     IGRILB(I)  y-dir grid box number of lower river boundary column I
C     IGRIUB(I)  y-dir grid box number of upper river boundary column I
C     LCSTSQ(I)  last section number of branch I
C     NSLSCT(I)  No. of slices of data for section I
C     NSTUBE(I)  No. of stream tubes for section I
C     NUMCON(I)  Condition Number (see text) for section I
C     NFIRCO(I)  Next section first connecting to section I
```

```
C     Q(I)        Discharge in the Ith Branch
C     SCTANG(I)   angle Ith section makes with X-direction
C     THETAO(I)   The clockwise angle(deg) of the Y-axis of the river I,
C                  measured from magnetic north.
C     WL(I)       water level of upstream of branch I
C     ZD(I)       reference level from datum for section I at which Z's
C                  are evaluated
C
C   -- Two D arrays
C     TICE(I,J)  Equivalent ice thickness of Jth vertical in Ith section
C     YWID(I,J)  Distance from lower bank of river to the Jth vertical
C                  in Ith section
C     Z(I,J)     Height of Jth vertical in Ith section
C
C   -- Complex Variables (these store X-component as real part and
C        Y-component as imaginary part)
C     CORDLB(I)  lower bank co-ords of the Ith section
C     CORDV(I,J) co-ords at which VSTRM(I,J) is acting
C     VSTRM(I,J) stream velocity of the Ith section and Jth streamtube
C     VCAR(I)    velocity in the cartesian box grid system of box I
C   ------------------------------------------------------------------
C
      READ(1,650)WORD,TEXT
      READ(1,*)NBRNCH,NGRIDX,DX,KINTM
      READ(1,*)(LCSTSQ(I),I=1,NBRNCH)
      IS2 = LCSTSQ(NBRNCH)+ 1
      DO 100 I=1,IS2
        READ(1,*)J,CORDLB(I),SCTANG(I),NSTUBE(I),NUMCON(I),NFIRCO(I)
     $  ,NSECO(I)
        IF(J.NE.I)WRITE(*,700)
C       SCTANG(I) = SCTANG(I)*3.141592/180.
100     CONTINUE
      DO 110 I=1,IS2
        READ(1,*)J,NSLSCT(I),ZD(I)
        IF(J.NE.I)WRITE(*,710)
        NNN=NSLSCT(I)+1
        READ(1,*)(YWID(I,J),Z(I,J),J=2,NNN)
110     CONTINUE
      DO 120 I=1,NGRIDX
        READ(1,*)J,IGRILB(I),IGRIUB(I),IGRLB1(I),IGRUB1(I)
        IF(I.NE.J)WRITE(*,720)
120     CONTINUE
C
C      Read the I,J values of Grid boxes in which velocity =0.0
C
      READ(1,*)NZRVB
      IF(NZRVB.EQ.0)GOTO 140
      IF(NZRVB.GT.100)WRITE(*,730)
      IF(NZRVB.GT.100)NZRVB=100
      READ(1,*)(IZRBX(I),IZRBY(I),I=1,NZRVB)
C
C    Read the spill volume, spill location and wind data
C
140     READ(12,650)FUELTP
        READ(12,*)TOTIME,IEVERY,IOPT1,IOPT2,IOPT3,IOPT4,SPLTIM,DIFFUD
```

```
      READ(12,*)NTOTAL,SPVOL,SPILDT,SPGOIL,ANIU,SIGMA,AK2I,AK2V,AK2T
     $ ,AKC10,AKC20,AKC30
      ISPTYP = 0
      IF(SPLTIM.GT.0.5*SPILDT)ISPTYP=1
      READ(14,*)ANICE,AMUNI
C
C     SPVOL IS U.S. GALLONS.  VOLPAR IS CU FT. OF VOLUME PER PARTICLE
C
      SPLRAT = 0.13368*SPVOL/SPLTIM
      VOLPAR = 0.13368*SPVOL/NTOTAL
      VZERO = SPVOL*3.7850E-03
      API = 0.0
      READ(12,*)SPX,SPY,VMUNI,SOLUNI,CEVP,T0EVP
      SOLBLT = SOLUNI*16018.453
      VMOL = VMUNI*0.02831682
      AMIUO = AMUNI*14.88162
      TOUNI = T0EVP*9./5.0
      IF(T0EVP.LT.1.0)API = 141.5/SPGOIL - 131.5
      APITEM = 141.5/SPGOIL -131.5
      SPCEN0 = CMPLX(SPX,SPY)
C
C     Check if the spill co-ordinates are in land
C     (This is a check for input error)
C
      L = SPX/DX + 1.0
      M = SPY/DX + 1.0
      IF(M.LT.IGRILB(L).OR.M.GT.IGRIUB(L))GOTO 150
      IF(IGRLB1(L).EQ.0)GOTO 160
      IF(M.GE.IGRLB1(L).AND.M.LE.IGRUB1(L))GOTO 150
      GOTO 160
150   WRITE(*,800)L,M
      STOP
160   CONTINUE
      TTTT=SPLTIM/60.
      IF(ISPTYP.EQ.1)WRITE(*,810)TEXT,SPCEN0,TTTT
      IF(ISPTYP.EQ.0)WRITE(*,820)TEXT,SPCEN0
      WRITE(*,830)TOTIME,NTOTAL,SPVOL,FUELTP,SPILDT,SPGOIL,APITEM
     $ ,ANIU,SIGMA
      WRITE(*,840)AK2I,AK2V,AK2T,AKC10,AKC20,AKC30,VMUNI,SOLUNI
      WRITE(*,842)AMUNI,ANICE
      IF(API.LT.1.) WRITE(*,844)CEVP,T0EVP
      IF(DIFFUD.LT.0.0)WRITE(*,846)
      IF(DIFFUD.GT.0.0)WRITE(*,847)DIFFUD
844   FORMAT(' API option is not selected . Evap. constants are ',
     $ ' C= ',F6.2,'  T0= ',F7.1)
846   FORMAT(//' Surface Diffusion - Default formulation is used'//)
847   FORMAT(//' Surface Diffusion Coeff. =',F6.2,' sq ft/sec'//)
C
C     READ BOUNDARY TYPE INFORMATION
C
170   READ(8,*)K,LFROM,LTO,ICODE
      IF(K.EQ.0)GOTO 190
      AN = HLIFE(ICODE)*3600./SPILDT
      REJRAT = 1 - 0.5**(1./AN)
      DO 180 L=LFROM,LTO
```

```
180   TYPBND(K,L)=REJRAT
      GOTO 170
190   CONTINUE
      NSPILS=(SPLTIM+1.0)/SPILDT
      IF(ISPTYP.EQ.1)NPERDT = NTOTAL/NSPILS
      IF(ISPTYP.EQ.1)GOTO 210
      NPERDT=NTOTAL
      DO 200 I=1,NTOTAL
200   PARTCL(I) = SPCENO
210   CONTINUE
C
C    First set Vol of each pie=8*one eighth of vol released in SPILDT
C
      DO 220 I=1,8
220   VOLPIE(I) = VOLPAR*NPERDT
C
C     set random number generation seed IX
C
      IX=101
      CALL RANDOMIZE(IX)
      WRITE(11,650)TEXT,FUELTP
      WRITE(11,651)NTOTAL,SPVOL,SPILDT,SPGOIL,ANIU,SIGMA,VMUNI,SOLUNI
     $   ,AMUNI
      TIMET = 0.
      IPARTX(1)=REAL(SPCENO)
      IPARTY(1)=AIMAG(SPCENO)
      WRITE(11,850)IPARTX(1),IPARTY(1)
      INDX1D = 0
      NST =1
      NPTCL = NPERDT
      FEVP1=0.
      FEVP2=0.
      TOTDIS=0.
      NBRP1 = NBRNCH + 1
      READ(7,*)UFDT
      WRITE(*,845)UFDT
      UFSTPS = TOTIME/UFDT
      OSTPS = (UFDT*3600.+1.0)/SPILDT
C
      DO 340 IUF=1,UFSTPS
C
C   Read Data Created by unsteady Flow Model
C
        IF(WORD.EQ.STCL)IRCODE=1
        IF(WORD.EQ.DETR)IRCODE=2
        IF(WORD.EQ.STMU)IRCODE=3
        IF(WORD.EQ.STML)IRCODE=4
C
C     If the numbering sequence of branches in oilspill model is the
C     same as that of 1-D model the following 3 statements can be used to
C     Read  the Q & WL data.  In this case subroutine NDCONV is not needed.
C     Subroutine NDCONV is specifically written for reading Q & WL from
C     the three 1-D River models St. Clair, Detroit and St. Mary's
C
C       DO 230 I=1,NBRP1
```

```
C           READ(7,*)WL(I),Q(I)
C230        CONTINUE
        CALL NDCONV(NBRP1,IRCODE)
C
C     Read ice thickness information
C
       READ(7,*)ICINFO
       DO 270 I=1,ICINFO
          READ(7,660)IS,WORD
          NNN=NSLSCT(IS)+1
           IF(ICINFO.EQ.1.AND.WORD.EQ.OPEN) THEN
           WRITE(*,234)
           ELSE
           IF(I.EQ.1)WRITE(*,235)
           ENDIF
          IF(WORD.NE.FULL)GOTO 250
          READ(7,*)FULLTI
          DO 240 K=1,NNN
             TICE(IS,K)=FULLTI
240          CONTINUE
          WRITE(*,236)IS,FULLTI
250          IF(WORD.EQ.PART) THEN
             READ(7,*)(TICE(IS,J),J=1,NNN)
             DO 252 J=1,NNN
             IDUM(J) = YWID(IS,J)
252          CONTINUE
             WRITE(*,237) IS,(IDUM(J),J=1,NNN)
             WRITE(*,238) (TICE(IS,J),J=1,NNN)
             ENDIF
          IF(WORD.NE.OPEN)GOTO 270
          DO 260 K=1,NNN
             TICE(IS,K)=0.0
260          CONTINUE
270    CONTINUE
C
       READ(14,*)NICERG
       IF(NICERG.EQ.0)GOTO 278
       DO 275 I=1,NICERG
         READ(14,*)NICEX1(I),NICEY1(I),NICEX2(I),NICEY2(I)
275      CONTINUE
       WRITE(*,930)NICERG,(I,NICEX1(I),NICEY1(I),NICEX2(I),NICEY2(I),
      $ I=1,NICERG)
C
C      Set up the 1D-array locations that define Ice regions
C
       DO 40 K=1,NICERG
         LK1=NICEX1(K)-1
         MK1=NICEY1(K)-1
         IPOS1(K) = 0
         IF(LK1.EQ.0)GOTO 45
         DO 44 L1=1,LK1
           IPOS1(K) = IPOS1(K)+IGRIUB(L1)-IGRILB(L1)+3
44         CONTINUE
45       IPOS1(K) = IPOS1(K)+MK1-IGRILB(LK1+1)+3
         LK2=NICEX2(K)-1
```

```
       MK2=NICEY2(K)-1
       IPOS2(K) = 0
       IF(LK2.EQ.0)GOTO 48
       DO 47 L1=1,LK2
         IPOS2(K) = IPOS2(K)+IGRIUB(L1)-IGRILB(L1)+3
47       CONTINUE
48     IPOS2(K) = IPOS2(K)+MK2-IGRILB(LK2+1)+3
40    CONTINUE
C
278   WRITE(*,860)
      DO 280 I=1,NBRNCH
280   WRITE(*,870)I,Q(I),WL(I)
C
C     Now call VELDIS to find the 2-D vel distribution in the river
C
      IF(IOPT1.EQ.1)CALL PRINT1(2,NBRNCH,NGRIDX,DX)
                    CALL VELDIS(IOPT2,NBRNCH,NGRIDX,DX)
C
C
        NST1=(IUF-1)*OSTPS+1
        NST2= NST1 + OSTPS -1
      DO 330 I=NST1,NST2
      READ(12,*)VWMAG,THETA,TENVF
      THET = (THETA0(IRCODE)-THETA)*3.141592/180.
      VWX=VWMAG*SIN(THET)
      VWY= - VWMAG*COS(THET)
      VWIND = CMPLX(VWX,VWY)
      WNDSPD = VWMAG/3.28
      VWMPH = VWMAG*0.6818
      TENV = (TENVF-32)*5./9. + 273.
      INDPRN = 0
      IF(MOD(I-1,IEVERY).EQ.0)INDPRN = 1
      TIMET = TIMET + SPILDT
      IF(ISPTYP.NE.1)GOTO 290
C
      IF(I.LE.NSPILS)CALL PRELSE(DX,SPILDT,IX,NST,NPTCL,SPCEN0,DIFFUD)
      IF(I.GT.1)CALL ADVECT(DX,SPILDT,IX,1,NST-1,DIFFUD)
290   IF(ISPTYP.EQ.0)CALL ADVECT(DX,SPILDT,IX,1,NTOTAL,DIFFUD)
      CALL ORIENT(INDX1D,DX)
C
      IF(INDX1D.LT.3)GOTO 293
      IF(NICERG.EQ.0)INDX1D=INDX1D-3
      IF(NICERG.EQ.0)GOTO 293
      NPTICE=0
      DO 292 KK=1,NMOVIN
        J= IMOVIN(KK)
        L = REAL(PARTCL(J))/DX
        M =(AIMAG(PARTCL(J))+YSHIFT(J))/DX
        IPOS = 0
        IF(L.EQ.0)GOTO 117
        DO 115 L1=1,L
          IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
115     CONTINUE
117     IPOS = IPOS+M-IGRILB(L+1)+3
        DO 118 K=1,NICERG
```

```
          IF(IPOS.GE.IPOS1(K).AND.IPOS.LE.IPOS2(K))NPTICE=NPTICE+1
118       CONTINUE
292     CONTINUE
        RATICE=FLOAT(NPTICE)/FLOAT(NMOVIN)
        IF(RATICE.LT.0.5)INDX1D=INDX1D-3
        IF(RATICE.GE.0.5)INDX1D=0
C
293   TTTT=TIMET/3600.
      GALPAR = VOLPAR/0.13368
      IF(INDPRN.EQ.1)WRITE(*,880)TTTT,VWMPH,THETA,TENVF,SPCEN,GALPAR,
     $ SLINFO(INDX1D+1)
      IF(INDX1D.EQ.0)CALL SPRDAX(DX,SPILDT,TIMET,INDPRN,SPAREA
     $ ,SPLTIM,SPLRAT)
      IF(INDX1D.EQ.1)CALL SPRD1Y(DX,SPILDT,TIMET,INDPRN,SPAREA)
      IF(INDX1D.EQ.2)CALL SPRD1X(DX,SPILDT,TIMET,INDPRN,SPAREA)
        FEVP1=FEVP2
      CALL EVAPOR(API,TENV,WNDSPD,VMOL,VZERO,SPAREA,SPILDT,I)
      CALL DISOLU(SPAREA,SOLBLT,TIMET,SPILDT,API,DELDIS,TOTDIS)
      DELUNI= DELDIS*264.172E-06/SPGOIL
      TOTUNI= TOTDIS*264.172E-06/SPGOIL
      VOLPAR= 0.13368*(SPVOL*(1-FEVP2)-TOTUNI)/NTOTAL
      IF(NHITB.GT.0)CALL BOUNDR(DX,NGRIDX,INDPRN)
C
      IF(I.GE.NSPILS)NST = NPTCL +1
      IF(I.GE.NSPILS)GOTO 300
      NST=NPTCL+1
      NPTCL=NST+NPERDT-1
C
300   IF(INDPRN.NE.1)GOTO 330
      WRITE(*,900)FEVP2,DELUNI,TOTUNI
      IF(IOPT4.EQ.1)CALL PLOTNU(DX)
      IF(IOPT3.NE.1)GOTO 330
      IPARTX(1)=REAL(SPCEN)
      IPARTY(1)=AIMAG(SPCEN)
      WRITE(11,910)IPARTX(1),IPARTY(1),TTTT,VWX,VWY,GALPAR
      DO 320 J=1,NTOTAL,5
         DO 310 K=1,5
            IPARTX(K) = REAL(PARTCL(J+K-1))
            IPARTY(K) = AIMAG(PARTCL(J+K-1))
310         CONTINUE
         WRITE(11,850)(IPARTX(K),IPARTY(K),K=1,5)
320      CONTINUE
330   CONTINUE
340   CONTINUE
      STOP
234   FORMAT(//' Open Water Conditions')
235   FORMAT(//' Ice conditions at cross sections',/1H+,32('_')//
     $' X-sect',9X,'Condition (A thickness of 0.0 implies open water)')
236     FORMAT(/I4,5X,'Ice cover of uniform thickness =',F5.2,
     $ ' ft across the river')
237   FORMAT(/I4,5X,'Partial or non-uniform ice cover across the river.'
     $,' Distances from'/9X,'the lower bank and corresponding ice'
     $ ' thickness is given below'/9X,'Dist(ft) ',20I5)
238   FORMAT(9X,'Thic(ft) ',20F5.2)
720   FORMAT(' ** ERROR ** READING GRID INFO.')
```

```
730   FORMAT(' NZRVB is GT 100 and is reset to 100')
710   FORMAT(' ** ERROR ** READING X SECTION - DATA ')
700   FORMAT(' ** ERROR ** READING LOWER BOUNDARY - DATA ')
660   FORMAT(I4,1X,A4)
650   FORMAT(20A4)
651   FORMAT(I4,F8.0,F7.0,F6.2,5E11.3)
810   FORMAT(1H1,16X,11A4,//5X,25('*')/5X,'*    CONTINUOUS SPILL       *'
     $ /5X,'*',11X,'AT',10X,'*'/5X,'*',4X,F7.0,',',F7.0,4X,'*'/
     $ 5X,'*',5X,'FOR ',F5.0,' min.',4X,'*'/5X,25('*'))
820   FORMAT(1H1,16X,11A4,//5X,25('*')/5X,'*  INSTANTANEOUS SPILL  *'
     $  /5X,
     $ '*',11X,'AT',10X,'*'/5X,'*',4X,F7.0,',',F7.0,4X,'*'/5X,25('*'))
830   FORMAT(//' SIMULATION PERIOD  = ',F5.1,' Hrs'///
     $ ' Characteristics of spill'/1H+,24('_')//
     $ ' No. of particles               :',I5,/
     $ ' Oil spilled                    :',F8.0,' gals of ',4A4/
     $ ' DT for spill simulation        :',F8.0,' Secs.',/
     $ ' Specific gravity of oil        :',F8.2,' (API index =',F5.1,')'/
     $ ' Kinematic Visco. of Water  :',E10.4,' sq ft/sec',/
     $ ' Surafce Tension                :',E10.4,' lbs/ft',/)
840   FORMAT(/9X,'Spreading Coefficients' /
     $ '  K2i    K2v    K2t    c10    c20    c30'/,6F6.2//
     $ ' Molar volume                   :',E10.4,' cu ft/mol'/
     $ ' Solubility of fresh oil     :',E10.4,' lbs/cu ft')
842   FORMAT (' Viscosity of Oil                :',F8.2,'lbs/ft-sec'/
     $ ' Manning s Roughness of Ice :',F8.3/)
845   FORMAT(/' Time step for river flow computation =',F6.2,' Hrs')
910   FORMAT(I8,I7,F9.4,2F6.1,F8.2)
850   FORMAT(5(I8,I7))
880   FORMAT(///1X,78('-')/' Time = ',F6.2,' Hrs -- Wind :mag=',F5.1,
     $ ' mph, dir =',F5.1,' deg    -- Air Temp=',F5.1,' F'/
     $ ' Spill center after advection= ',F7.0,',',F7.0  '(ft)'/
     $ ' Volume per particle            = ',F8.2,' gals'//
     $ '        Slick Condition during this time step'//
     $ ' Slick information by pie / strip'/A44)
900   FORMAT(//' Slick condition at the end of this time step'
     $ //' Fraction Evaporated = ',G10.5/' Amount Dissolved (gals)
     $ : This Step = ',G10.5,' Total = ',G10.5/)
860   FORMAT(//' Flow and Discharge Conditions in River'/
     $ '  Branch    Q (cfs)    Stage (ft)    ')
870   FORMAT(4X,I2,5X,F7.0,5X,F6.2)
800   FORMAT(//'  Spill location co-ords are in land X & Y GRID box no. s
     $ are ',I4,' &',I4,//' ^^ Execution terminated ^^')
930   FORMAT(/20X,'No. of Ice Covered Regions =',I3/' Region    from'
     $ ' X,Y Grid   to X,Y Grid'/(I4,11X,I3,',',I3,6X,I3,',',I3))
      END
```

Subroutine ADVECT

```
      SUBROUTINE ADVECT(DX,SPILDT,IX,N1,N2,DIFFUD)
C
C     This subroutine handles the advection of spill, in each time step
C     Each particle is advected according to current & wind velociites
C     (see text for details)
C     This routine advects moving particles in the range N1 to N2
C     This version includes advection under ice covers
C
C     Last date of revision :  Jun 25, 1986
C
      COMPLEX VCAR(12000),SPCEN,PARTCL(1000),VWIND,VDRIFT
      COMMON /VA/ VCAR,VWIND,VDRIFT
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      COMMON /BLOCK7/SPGOIL,ANIU,SIGMA,AK2I,AK2V,AK2T,
     $ VOLPAR,VOLPIE(8),SLICKR(8)
      COMMON /ICE/NICEX1(20),NICEY1(20),NICEX2(20),NICEY2(20),NICERG,
     $ AMIUO,ANICE,IPOS1(20),IPOS2(20),SPAICE
C
C     Input : .. Location of each particle
C             .. Velocity distribution in river
C     Output: .. New loaction of each particle
C
      DATA STDEVI/0.050/,PI /3.141592/
      IF(NICERG.EQ.0)GOTO 25
C
C     DELEQ - Equilibrium thickness (ft)
C     UTH   - Threshold current speed for slick movement ( ft/sec)
C     UFAIL - Failure velocity under rough ice cover ( ft/sec)
C     FRAMFA- FRiction AMplification FActor denoted by 'K' in text
C     AMIUO - Viscosity of Oil in g/cm-sec
C
      DELEQ = (1.67 - 8.5*(1.0-SPGOIL))/30.48
      UTH = (205.79/(88.68-AMIUO))/30.48
      FRAMFA = 35.55*ANICE + 1.0
      IF(ANICE.GT.0.045)FRAMFA = 2.6
      TERM1=SQRT(SIGMA*(32.2)**2*62.4*(1.-SPGOIL))
      UFAIL=1.5*SQRT(2.*(1.+SPGOIL)*TERM1/(62.4*SPGOIL))
C
C     loop 60 operates for each moving particle in the system
C
25    DO 60 I=N1,N2
         SUMDT = 0.
         IPASS = 1
C     SUMDT - Sum of the samll Dt's (DTSMALL)
C     IPASS - pass number in this loop.  A prticle may move from its
C     previous position to present position through only one pass or
C several passes depending on the magnitudes of velocities in the region
40       DO 30 J=1,NHITB
         IF(I.EQ.IHITB(J))GOTO 60
30      CONTINUE
        L = REAL(PARTCL(I))/DX
        M =AIMAG(PARTCL(I))/DX
        IPOS = 0
```

```
          IF(L.EQ.0)GOTO 117
          DO 115 L1=1,L
           IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
115       CONTINUE
117       IPOS = IPOS+M-IGRILB(L+1)+3
          IF(NICERG.EQ.0)GOTO 125
C
C       determine whether the particle is under ice or not
C
          ICOND=0
          DO 120 K=1,NICERG
            IF(IPOS.GE.IPOS1(K).AND.IPOS.LE.IPOS2(K))ICOND=1
            IF(ICOND.EQ.1)GOTO 180
120         CONTINUE
C
C       Advection velocity in free-surface conditions
C
125     VDRIFT = 0.03*VWIND + 1.1*VCAR(IPOS)
        GOTO 210
C
C       Advection Velocity under Ice
C
180     UWATER = CABS(VCAR(IPOS))
        ROUGH = (ANICE/0.034)**6
        IF(ROUGH.GT.DELEQ)GOTO 190
        IF(UWATER.LT.UTH)GOTO 60
        GOTO 200
190     IF(UWATER.LT.UFAIL)GOTO 60
200     VDRIFT = VCAR(IPOS)
        FDELTA = UWATER/SQRT((1.0-SPGOIL)*32.2*DELEQ)
        FK = SQRT(FRAMFA/(0.115*FDELTA**2 + 1.105))
        VDRIFT = VDRIFT*(1-FK)
210     DTSMAL = 86400.
C       86400 is just a large number in this case equal to secs in a day
        VELUPV = ABS(REAL(VDRIFT)) + ABS(AIMAG(VDRIFT))
        IF(VELUPV.GT.0.1)DTSMAL = DX/VELUPV
        IF(DTSMAL.GT.SPILDT.AND.IPASS.EQ.1)DTSMAL = SPILDT
        IF(DTSMAL.GT.SPILDT.AND.IPASS.GT.1)DTSMAL = SPILDT - SUMDT
        IF(DTSMAL.LT.0.0)DTSMAL = 0.
        IF((SUMDT+DTSMAL).GE.SPILDT)DTSMAL = SPILDT - SUMDT
        IPASS = IPASS + 1
        IF((SUMDT+DTSMAL).GE.SPILDT) IPASS = 9999
        SUMDT = SUMDT + DTSMAL
        CALL RANDU(IX,IY,YFL)
        IX = IY
        ANG = PI*YFL
        CALL GAUSS(IX,1.0,0.0,VRAND)
        IF (DIFFUD.LT.0.0)DDD = 2.88*CABS(VDRIFT)
        IF (DIFFUD.GE.0.0)DDD = 4*DIFFUD
        VRAND = VRAND*SQRT(DDD/DTSMAL)
        VX = VRAND*COS(ANG)
        VY = VRAND*SIN(ANG)
c
        STDEV = 100.*SQRT(DDD/DTSMAL)/CABS(VDRIFT)
        VMAG = CABS(VDRIFT)
```

```
      VDRIFT = VDRIFT+ CMPLX(VX,VY)
      PARTCL(I) = PARTCL(I) + DTSMAL*VDRIFT
      STDEV = 100.*VRAND / CABS(VDRIFT)
      IF(IPASS.NE.9999)GOTO   40
C
C     Check for particle hitting the boundaries
C
      L = REAL(PARTCL(I))/DX + 1
      M = AIMAG(PARTCL(I))/DX + 1
      IF(M.GE.IGRILB(L).AND.M.LE.IGRIUB(L))GOTO 55
      NHITB = NHITB + 1
      IHITB(NHITB) = I
55    IF(IGRLB1(L).EQ.0)GOTO 59
      IF(M.LE.IGRUB1(L).AND.M.GE.IGRLB1(L))GOTO 58
      GOTO 59
58    NHITB = NHITB+1
      IHITB(NHITB) = I
59       CONTINUE
      IF(IPASS.NE.9999)GOTO 40
60    CONTINUE
      RETURN
      END
```

Subroutine BOUNDR

```
      SUBROUTINE BOUNDR(DX,NGRIDX,INDPRN)
      COMPLEX SPCEN,PARTCL(1000)
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      DIMENSION NPTBND(4,300),IDUM1(300),IDUM2(300)
C
C     This subroutine handles adsorption and rejection at the river
C     boundaries
C
C     Last date of revision March 04,1985
C
      DO 10 I=1,NGRIDX
        DO 10 K=1,4
10      NPTBND(K,I)=0
      DO 80 I=1,NHITB
      J = IHITB(I)
      L = REAL(PARTCL(J))/DX + 1
      M = AIMAG(PARTCL(J))/DX + 1
C
C     Check if the particle is below the lower boundary, If so assign it
C     it to the boundary grid and count
C
      IF(M.GE.IGRILB(L))GOTO 20
      IF(M.EQ.(IGRILB(L)-1))GOTO 15
      X1 = REAL(PARTCL(J))
      Y1 = IGRILB(L)*DX - 1.5*DX
      PARTCL(J) = CMPLX(X1,Y1)
15    NPTBND(1,L) = NPTBND(1,L) + 1
      GOTO 80
C
C     Check if the particle is above the upper boundary, If so assign it
C     it to the boundary grid and count
C
20    IF(M.LE.IGRIUB(L))GOTO 40
      IF(M.EQ.(IGRIUB(L)+1))GOTO 30
      X1 = REAL(PARTCL(J))
      Y1 = IGRIUB(L)*DX + DX/2.
      PARTCL(J) = CMPLX(X1,Y1)
30    NPTBND(2,L) = NPTBND(2,L) + 1
      GOTO 80
C
C     If it didn't belong to the two categories above it must be in the
C     Island, therefore assign to the nearest boundary, and count.
C
40    Y2 = AIMAG(PARTCL(J))-IGRLB1(L)*DX + 0.75*DX
      Y3 = IGRUB1(L)*DX - 0.25*DX - AIMAG(PARTCL(J))
      IF(Y2.LT.Y3)PARTCL(J) = PARTCL(J) - CMPLX(0.,Y2)
      IF(Y2.LT.Y3)NPTBND(3,L) = NPTBND(3,L) + 1
      IF(Y3.LT.Y2)PARTCL(J) = PARTCL(J) + CMPLX(0.,Y3)
      IF(Y3.LT.Y2)NPTBND(4,L) = NPTBND(4,L) + 1
80    CONTINUE
C
C     No. of particles in each boundary grid has been determined
C     Now check for the boundary type and re-entrain the excess particles
```

```
C
      IF(INDPRN.EQ.1)WRITE(*,300)
      DO 220 L1=1,NGRIDX
      NBNDR = 2
      IF(IGRLB1(L1).NE.0)NBNDR=4
      DO 210 K=1,NBNDR
      IF(NPTBND(K,L1).EQ.0)GOTO 210
      IALOWD = 0.5 +  (1.-TYPBND(K,L1))*NPTBND(K,L1)
C     IF(INDPRN.EQ.1)WRITE(*,310)K,L1,NPTBND(K,L1)
      KOUNT = 0
      I = 0
90    I = I + 1
      IF(I.GT.NHITB)GOTO 205
      J = IHITB(I)
      L = REAL(PARTCL(J))/DX + 1
      IF(L.NE.L1)GOTO 90
      M = AIMAG(PARTCL(J))/DX + 1
      IF(K.NE.1)GOTO 110
      IF(M.NE.IGRILB(L)-1)GOTO 90
      KOUNT = KOUNT + 1
      IF(KOUNT.LE.IALOWD)GOTO 90
      XCO=L*DX - 0.5*DX
      YCO=M*DX + 0.5*DX
      PARTCL(J) = CMPLX(XCO,YCO)
      NHITB = NHITB - 1
      DO 105 II =I,NHITB
105   IHITB(II) = IHITB(II+1)
      IHITB(NHITB+1)=0
          I=I-1
          GOTO 90
C
110   IF(K.NE.2)GOTO 130
      IF(M.NE.IGRIUB(L)+1)GOTO 90
      KOUNT = KOUNT + 1
      IF(KOUNT.LE.IALOWD)GOTO 90
      XCO=L*DX - 0.5*DX
      YCO=M*DX - 1.5*DX
      PARTCL(J) = CMPLX(XCO,YCO)
      NHITB = NHITB - 1
      DO 115 II =I,NHITB
115   IHITB(II) = IHITB(II+1)
      IHITB(NHITB+1)=0
        I=I-1
        GOTO 90
C
130   IF(NBNDR.EQ.2)GOTO 90
      IF(K.NE.3)GOTO 150
      IF(M.NE.IGRLB1(L))GOTO 90
      IF(IGRUB1(L).NE.IGRLB1(L))GOTO 140
      XXX = AIMAG(PARTCL(J))-(IGRLB1(L)-1)*DX
      IF(XXX.GT.0.5*DX)GOTO 90
140   KOUNT = KOUNT + 1
      IF(KOUNT.LE.IALOWD)GOTO 90
      XCO=L*DX - 0.5*DX
      YCO=M*DX - 1.5*DX
```

```
      PARTCL(J) = CMPLX(XCO,YCO)
      NHITB = NHITB - 1
      DO 125 II =I,NHITB
125   IHITB(II) = IHITB(II+1)
      IHITB(NHITB+1)=0
         I=I-1
         GOTO 90
C
150   IF(K.NE.4)GOTO 90
      IF(M.NE.IGRUB1(L))GOTO 90
      IF(IGRUB1(L).NE.IGRLB1(L))GOTO 160
      XXX = IGRUB1(L)*DX - AIMAG(PARTCL(J))
      IF(XXX.GT.0.5*DX)GOTO 90
160   KOUNT = KOUNT + 1
      IF(KOUNT.LE.IALOWD)GOTO 90
      XCO=L*DX - 0.5*DX
      YCO=M*DX + 0.5*DX
      PARTCL(J) = CMPLX(XCO,YCO)
      NHITB = NHITB - 1
      DO 135 II =I,NHITB
135   IHITB(II) = IHITB(II+1)
      IHITB(NHITB+1)=0
         I=I-1
      GOTO 90
205      CONTINUE
210   CONTINUE
220   CONTINUE .
      IF(INDPRN.EQ.0)RETURN
      DO 430 K=1,4
      KOUNT=0
      DO 410 L=1,NGRIDX
      IF(NPTBND(K,L).EQ.0)GOTO 410
      KOUNT = KOUNT+1
      IDUM1(KOUNT)=L
      IDUM2(KOUNT)=NPTBND(K,L)
      IF(KOUNT.GT.250)WRITE(420)
410   CONTINUE
      IF(KOUNT.EQ.0)GOTO 430
      K1=1
      K2=KOUNT
      IF(KOUNT.GT.28)K2=28
415   WRITE(*,440)K,(IDUM1(I),I=K1,K2)
      WRITE(*,450)(IDUM2(I),I=K1,K2)
      IF(K2.GE.KOUNT)GOTO 430
      K1=K2+1
      K2=K1+27
      GOTO 415
430   CONTINUE
      RETURN
300   FORMAT(/25X,'Oil in River Banks')
440   FORMAT(/' Bank',I2,'; X-Grid',28I4)
450   FORMAT(6X,'Particles',28I4)
310   FORMAT(5X,3(I3,3X))
      END
```

Subroutine DISOLU

```
      SUBROUTINE DISOLU(SPAREA,SOLBLT,TIMET,SPILDT,API,DELDIS,TOTDIS)
C
C     This subroutine computes the amount of oil dissolved in water
C     The solubility of oil is so low that it does not affect the
C     trajectory (spreading), but is important for environmental impact
C     assesment. --- The working units in this subroutine is METRIC
C
C     Last date of Revision : October 15, 1985
C
C     Explanation of variables
C     DISOLK  - Dissolution mass transfer coefficient    1 cm/hr
C                                              or 2.7777E-06 m/sec
C     SOLBLT  - Solubility of fresh oil (g/cu m)
C     ARBAR   - mean area of slick during the time step (sq. m)
C     DELDIS  - amount of oil dissolved during time step (grams)
C     SPAREA  - Free surface area of spill (sq. ft)
C     SPAICE  - Area of spill under ice (sq. ft)
C
      COMMON /ICE/NICEX1(20),NICEY1(20),NICEX2(20),NICEY2(20),NICERG,
     $ AMIUO,ANICE,IPOS1(20),IPOS2(20),SPAICE
      DATA DISOLK/2.7777E-06/
      SPAR2=(SPAREA+SPAICE)/10.76
      ARBAR=(SPAR1+SPAR2)/2.0
      SPAR1=SPAR2
      T1=(TIMET-SPILDT)/36000
      T2=TIMET/36000.
      DELDIS=-DISOLK*ARBAR*SOLBLT*36E3*(EXP(-T2)-EXP(-T1))
      TOTDIS=TOTDIS+DELDIS
      RETURN
      END
```

Subroutine EVAPOR

```
      SUBROUTINE EVAPOR(API,TENV,WNDSPD,VMOL,VZERO,SPAREA,SPILDT,JSTEP)
C
C     This subroutine computes evaporatioon rates based on
C     Mackay, Patterson and Nadeau ' s theory.
C     In this subroutine metric unit system is used.  The reason for
C     for using units differnt units from other subroutine is to make
C     cross reference with theory easier.
C
C     Last date of revision October 29,1985
C
C     Explanation of variables used in EVAPOR
C     AKM     - mass transfer coeff. Km   (m/s)
C     P0      - vapor pressure at TENV    (atm)
C     C       - coefficient C at TENV
C     FEVP2       - fraction evaporated
C     JSTEP   - current time step
C     RGAS    - gas constant:the values of RGAS in differnt units are
C             -  1.98726  cal/deg mole
C             -  8.31470  joules/deg mole
C             -  82.0597  cc-atm/deg mole
C     SPAREA - Free surface area of spill (sq. ft)
C     SPAICE - Area of spill under ice (sq. ft)
C
      COMMON /BLOCK7/SPGOIL,ANIU,SIGMA,AK2I,AK2V,AK2T,
     $ VOLPAR,VOLPIE(8),SLICKR(8)
      COMMON /SE/FEVP1,FEVP2,CEVP,T0EVP
      DATA RGAS/8.3147/
      DATA SCKET,SUMC,SUMP0/3*0./
       SPAR2=SPAREA/10.76
       ARBAR=(SPAR2+SPAR1)/2.0
       SPAR1=SPAR2
       IF(WNDSPD.LT.0.0001)WNDSPD=0.001
      AKM = 0.0025*WNDSPD**0.78
          C=CEVP
          T0=T0EVP
          IF(API.LT.1.0)GOTO 50
      C = 1158.9*API**(-1.1435)
      T0=542.6-30.27*API+1.565*API**2-3.439E-02*API**3+2.604E-04*API**4
 50   P0= EXP(10.6*(1-T0/TENV))
      CC= C*283./TENV
      SUMP0 = SUMP0 + P0
      SUMC  = SUMC + CC
      P0BAR = SUMP0/JSTEP
      CBAR  = SUMC /JSTEP
      TOIL = TENV
      AKE = AKM*ARBAR*VMOL/(RGAS*TOIL*VZERO)
      SCKET = SCKET + CBAR*AKE*SPILDT*10.137E04
      SUME = SUME + AKE*SPILDT*10.137E04
      FEVP2 = (ALOG(P0BAR) + ALOG(SCKET+1./P0BAR))/CBAR
      IF(FEVP2.GT.0.6) FEVP2=0.6
      RETURN
      END
```

```
      SUBROUTINE NDCONV(NBRP1,IRCODE)
C     This subroutine reads data on water level and discharge according
C     to the sequence of ONE-D flow model and then converts to the
C     sequence required by Oilspill model.  If both have the same
C     sequence of numbering this subroutune is not required.
C     This subroutine is specifically for three rivers for which already
C     a 1-D model is availble
C               Code          River
C                 1             St. Clair
C                 2             Detroit
C                 3             Upper St. mary's
C                 4             Lower St. Mary's
C
C     Last date of revision : August 13, 1986
C
      DIMENSION DWL(22),DQ(22),NPTS(4)
      INTEGER RIV1(16),RIV2(16),RIV3(16),RIV4(16)
      COMPLEX VSTRM(99,16),CORDV(99,16),CORDLB(99)
      COMPLEX VWIND,VDRIFT
      COMMON /VEL/VSTRM,CORDV,CORDLB,Q(30),WL(30),TICE(99,20),
     $ YWID(99,20),Z(99,20),ZD(99),NSLSCT(99),SCTANG(99),
     $ LCSTSQ(30),NSTUBE(99),NUMCON(99),NFIRCO(99),NSECO(99),KINTM
      DATA RIV1/1,2,3,4,5,5,7,6,7,8,9,5*1/
      DATA RIV2/1,14,3,16,4,6,7,8,9,10,11,18,13,20,21,22/
      DATA RIV3/1,2,3,4,5,6,1,9*1/
      DATA RIV4/1,9,10,3,4,5,6,7,11,13,13,5*1/
      DATA NPTS/9,22,6,13/
      N=NPTS(IRCODE)
      DO 10 I=1,N
        READ(7,*)DWL(I),DQ(I)
10      CONTINUE
      DO 20 I=1,NBRP1
        IF(IRCODE.EQ.1)K=RIV1(I)
        IF(IRCODE.EQ.2)K=RIV2(I)
        IF(IRCODE.EQ.3)K=RIV3(I)
        IF(IRCODE.EQ.4)K=RIV4(I)
        WL(I)=DWL(K)
        Q(I)=DQ(K)
20      CONTINUE
      IF(IRCODE.EQ.2)Q(13)=Q(12)+Q(11)
      IF(IRCODE.NE.1)RETURN
      WL(7)=DWL(6)+ (DWL(5)-DWL(6))*20630./35680.
      Q(5) = DQ(6)*0.7
      Q(6) = DQ(6)*0.3
      RETURN
      END
```

Subroutine ORIENT

```
      SUBROUTINE ORIENT(INDX1D,DX)
C
C     This program computes the Orientation of the oil slick
C     and Aspect ratio.
C     If ASPECT >3  The Slick will be treated as 1-D.
C     Last date of Revision : Oct 17,1985
C
C     INDX1D=0   :  Axi-symmetrical spreading
C     INDX1D=1   : One-D spreading in Y-dir use SPRD1Y
C     INDX1D=2   : One-D spreading in X-dir use SPRD1X
C     INDX1D=3   :  Axi-symmetrical spreading(Short slick)
C     INDX1D=4   : One-D spreading in Y-dir use SPRD1Y (Short slick)
C     INDX1D=5   : One-D spreading in X-dir use SPRD1X (Short  slick)
C
      COMPLEX SPCEN,PARTCL(1000)
      COMMON /SO/IMOVIN(1000),YSHIFT(1000),NMOVIN,SSHIFT
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      DATA PI/3.141592/
      NMOVIN=0
      INDX1D = 0
      COUNT = 0.
      SPCEN = (0.,0.)
C
C     Find the indeces of moving particles and assign them to
C     array IMOVIN().  Also compute the spill center (SPCEN)
C
      DO 30 I=1,NPTCL
        DO 15 J=1,NHITB
          IF(I.EQ.IHITB(J))GOTO 30
15        CONTINUE
        NMOVIN=NMOVIN+1
        IMOVIN(NMOVIN)=I
        SPCEN = SPCEN + PARTCL(I)
        COUNT = COUNT + 1.
30      CONTINUE
      SPCEN = SPCEN/COUNT
C
C   If there is an island, any particles in the Northern Island are
C   shifted in (-)y-dir by a distance = width of island at that point.
C   IMPORTANT : The particles are moved back by equal amounts in
C   SPRDAX, SPRD1X or SPRD1Y subroutine.
C
      SSHIFT = 0.
      DO 430 I=1,NMOVIN
        J=IMOVIN(I)
        YSHIFT(J)=0.
        L=REAL(PARTCL(J))/DX+1
        IF(IGRLB1(L).EQ.0)GOTO 430
        M=AIMAG(PARTCL(J))/DX + 1
        IF(M.LE.IGRUB1(L))GOTO 430
        YSHIFT(J)=(IGRUB1(L)-IGRLB1(L)+1)*DX
        PARTCL(J)=PARTCL(J)-CMPLX(0.,YSHIFT(J))
        SSHIFT=SSHIFT+YSHIFT(J)
```

```
430     CONTINUE
C
C   If particles are shifted, re-Compute the Spill-Center
C
        SPX=REAL(SPCEN)
        SPY=AIMAG(SPCEN)
        IF(SSHIFT.LT.DX)GOTO 450
        SPY=SPY - SSHIFT/NMOVIN
        SPCEN = CMPLX(SPX,SPY)
450     SUMIX=0.
        SUMIY=0.
        SUMIXY = 0.
        AVGRAD=0.0
        SPX=REAL(SPCEN)
        SPY=AIMAG(SPCEN)
        DO 50 I=1,NMOVIN
          J=IMOVIN(I)
          XX=REAL(PARTCL(J))-SPX
          YY=AIMAG(PARTCL(J))-SPY
          AVGRAD = AVGRAD + SQRT(XX*XX+YY*YY)
          SUMIXY = SUMIXY + XX*YY
          SUMIY=SUMIY+ XX*XX
          SUMIX=SUMIX+ YY*YY
50        CONTINUE
        AVGRAD = AVGRAD/NMOVIN
        TOP= -2*SUMIXY
        BOT= SUMIX-SUMIY
        THETA=ATAN2(TOP,BOT)
        THETA=THETA/2.0
        IF(THETA.LT.0.0)THETA=THETA+2*PI
        CTHETA = COS(THETA)
        STHETA = SIN(THETA)
        SALONG=0.
        SNORML=0.
        DO 60 I=1,NMOVIN
          J=IMOVIN(I)
          XX=REAL(PARTCL(J))-SPX
          YY=AIMAG(PARTCL(J))-SPY
          SALONG = SALONG + ABS(XX*CTHETA+YY*STHETA)
          SNORML = SNORML + ABS(YY*CTHETA-XX*STHETA)
60        CONTINUE
        SALONG = SALONG/NMOVIN
        SNORML = SNORML/NMOVIN
        ASPECT = SALONG/SNORML
        IF(ASPECT.LT.1.0)THETA = THETA + 0.5*PI
        IF(ASPECT.LT.1.0)ASPECT = SNORML/SALONG
        IF(THETA.GT.2*PI)THETA=THETA - 2*PI
        IF(ASPECT.LT.3.0)GOTO 80
        INDX1D = 1
        IF(THETA.GT.0.25*PI.AND.THETA.LT.0.75*PI)INDX1D=2
        IF(THETA.GT.1.25*PI.AND.THETA.LT.1.75*PI)INDX1D=2
80      DEG= THETA*180./PI
        IF(AVGRAD.LT.0.5*DX)INDX1D = INDX1D+3
        RETURN
        END
```

Subroutine PLOTNU

```
      SUBROUTINE PLOTNU(DX)
C
C    This subroutine plots oil concentrations as the no. of particles
C    in each grid defined by size DX
C
C     Last date of revision : Apr 04, 86
C
      COMPLEX  SPCEN,PARTCL(1000),SPCEN1
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      COMMON /SO/IMOVIN(1000),YSHIFT(1000),NMOVIN,SSHIFT
      DIMENSION KOUNT(20,20)
      XXX=REAL(SPCEN)
      YYY=AIMAG(SPCEN)+SSHIFT/NMOVIN
      SPCEN1 = CMPLX(XXX,YYY)
C
C     Set all array elements such that they print stars on output
C
      DO 40 I=1,20
      DO 40 J=1,20,2
       KOUNT(I,J)=1001
       KOUNT(I,J+1)=1001
40     CONTINUE
      IMIN = REAL(SPCEN1)/DX - 9
      XMIN1=(IMIN- 1 )*DX
      IMAX= IMIN + 19
      JMIN=AIMAG(SPCEN1)/DX - 9
      YMIN1=(JMIN- 1 )*DX
      JMAX= JMIN + 19
      DO 80 I=1,20
         L=IMIN+I - 1
         IF(L.LT.1)GOTO 80
         M1 = IGRILB(L) - JMIN
         M2 = IGRIUB(L) - JMIN + 2
         IF(M1.LT.1)M1=1
         IF(M2.GT.20)M2=20
C
C   Now set array elements that corresponds to river+boundary to zero
C
         DO 70 J=M1,M2
            KOUNT(I,J)=0
70          CONTINUE
         IF(IGRLB1(L).EQ.0)GOTO 80
C
C    This part is for setting array   ments to print stars for Island
C
         M1 = IGRLB1(L) - JMIN +2
         M2 = IGRUB1(L) - JMIN
         IF(M1.LT.1)M1=1
         IF(M2.GT.20)M2=20
         IF(M1.GT.M2) GOTO 80
         DO 75 J=M1,M2
            KOUNT(I,J)=1001
75          CONTINUE
```

```
80      CONTINUE
C
C   Now check and count the no. of particles in the grid boxes
C
       DO 450 I=1,NPTCL
          L = (REAL(PARTCL(I)) -XMIN1)/DX+1
          M = (AIMAG(PARTCL(I))-YMIN1)/DX+1
          IF(L.LT.1.OR.L.GT.20) GOTO 450
          IF(M.LT.1.OR.M.GT.20) GOTO 450
          KOUNT(L,M)=KOUNT(L,M)+1
450       CONTINUE
       WRITE(*,620)JMIN,JMAX
       WRITE(*,610) (KOUNT(1,M),M=1,20),IMIN
       DO 580 L=2,19
          WRITE(*,600) (KOUNT(L,M),M=1,20)
580       CONTINUE
       WRITE(*,610) (KOUNT(20,M),M=1,20),IMAX
       RETURN
600    FORMAT(1X,20I3)
610    FORMAT(1X,20I3,' - X Grid#=',I3)
620    FORMAT(//'  Y Grid#=',I3,T56,I3,'=Y Grid#'/'   I',T60,'I')
       END
```

```
      SUBROUTINE PRELSE(DX,SPILDT,IX,N1,N2,SPCEN0,DIFFUD)
C
C     This subroutine, to be used for continuous spills, releases
C     particles (No.s N1 to N2) at SPCEN0.  Note that the No. of
C     particles released in SPILDT is NPERDT. Therefore NPERDT=N2-N1+1
C     The release will be at equal intervals of time.
C     This version30 has a modified advection term
c     this is version31 with modified diff. coef.
C
C     Last date of revision : Jun 26, 1986
C
      COMPLEX VCAR(12000),SPCEN,PARTCL(1000),VWIND,VDRIFT
      COMPLEX SPCEN0,VDR1
      COMMON /VA/ VCAR,VWIND,VDRIFT
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      COMMON /BLOCK7/SPGOIL,ANIU,SIGMA,AK2I,AK2V,AK2T,
     $ VOLPAR,VOLPIE(8),SLICKR(8)
      COMMON /ICE/NICEX1(20),NICEY1(20),NICEX2(20),NICEY2(20),NICERG,
     $ AMIUO,ANICE,IPOS1(20),IPOS2(20),SPAICE
C
C     Input : .. Location of spill center
C             .. Velocity distribution in river
C     Output: .. New location of each particle
C
      DATA PI /3.141592/
      IF(NICERG.EQ.0)GOTO 25
C
C     DELEQ - Equilibrium thickness (ft)
C     UTH   - Threshold current speed for slick movement ( ft/sec)
C     UFAIL - Failure velocity under rough ice cover ( ft/sec)
C     FRAMFA- FRiction AMplification FActor denoted by 'K' in text
C     AMIUO - Viscosity of Oil in g/cm-sec
C
      DELEQ = (1.67 - 8.5*(1.0-SPGOIL))/30.48
      UTH = (305.79/(88.68-AMIUO))/30.48
      FRAMFA = 35.55*ANICE + 1.0
      IF(ANICE.GT.0.045)FRAMFA = 2.6
      TERM1=SQRT(SIGMA*(32.2)**2*62.4*(1.-SPGOIL))
      UFAIL=1.5*SQRT(2.*(1.+SPGOIL)*TERM1/(62.4*SPGOIL))
      ROUGH = (ANICE/0.034)**6
C
25      L = REAL(SPCEN0)/DX
        M =AIMAG(SPCEN0)/DX
        IPOS = 0
        IF(L.EQ.0)GOTO 117
        DO 115 L1=1,L
         IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
115     CONTINUE
117     IPOS = IPOS+M-IGRILB(L+1)+3
        IF(NICERG.EQ.0)GOTO 125
C
C     determine whether the spill center is under ice or not
C
```

```
        ICOND=0
        DO 120 K=1,NICERG
          IF(IPOS.GE.IPOS1(K).AND.IPOS.LE.IPOS2(K))ICOND=1
          IF(ICOND.EQ.1)GOTO 180
120       CONTINUE
C
C     Advection velocity in free-surface conditions
C
125   VDRIFT = 0.03*VWIND + 1.1*VCAR(IPOS)
      GOTO 220
C
C     Advection Velocity under Ice
C
180     VDRIFT =(0.0,0.0)
      UWATER = CABS(VCAR(IPOS))
      IF(ROUGH.GT.DELEQ)GOTO 190
      IF(UWATER.LT.UTH)GOTO 220
      GOTO 200
190   IF(UWATER.LT.UFAIL)GOTO 220
200   VDRIFT = VCAR(IPOS)
      FDELTA = UWATER/SQRT((1.0-SPGOIL)*32.2*DELEQ)
      FK = SQRT(FRAMFA/(0.115*FDELTA**2 + 1.105))
      VDRIFT = VDRIFT*(1-FK)
C
220   VDR1 = VDRIFT
      DO 60 I=N1,N2
         DTPTCL = SPILDT*(I-N1+1)/(N2-N1+1)
         SUMDT = 0.
         IPASS = 1
         PARTCL(I) = SPCEN0
         VDRIFT = VDR1
C
40       IF(IPASS.EQ.1)GOTO 28
         IF(NHITB.EQ.0)GOTO 35
            DO 30 J=1,NHITB
               IF(I.EQ.IHITB(J))GOTO 60
30             CONTINUE
35      L = REAL(PARTCL(I))/DX
        M =AIMAG(PARTCL(I))/DX
        IPOS = 0
        IF(L.EQ.0)GOTO 517
        DO 515 L1=1,L
         IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
515     CONTINUE
517     IPOS = IPOS+M-IGRILB(L+1)+3
        IF(NICERG.EQ.0)GOTO 525
C
C     determine whether the spill center is under ice or not
C
        ICOND=0
        DO 520 K=1,NICERG
          IF(IPOS.GE.IPOS1(K).AND.IPOS.LE.IPOS2(K))ICOND=1
          IF(ICOND.EQ.1)GOTO 580
520       CONTINUE
C
```

```
C     Advection velocity in free-surface conditions
C
525   VDRIFT = 0.03*VWIND + 1.1*VCAR(IPOS)
      GOTO 620
C
C     Advection Velocity under Ice
C
580     VDRIFT = (0.0,0.0)
      UWATER = CABS(VCAR(IPOS))
      IF(ROUGH.GT.DELEQ)GOTO 590
      IF(UWATER.LT.UTH)GOTO 620
      GOTO 600
590   IF(UWATER.LT.UFAIL)GOTO 620
600   VDRIFT = VCAR(IPOS)
      FDELTA = UWATER/SQRT((1.0-SPGOIL)*32.2*DELEQ)
      FK = SQRT(FRAMFA/(0.115*FDELTA**2 + 1.105))
      VDRIFT = VDRIFT*(1-FK)
620   CONTINUE
C
28     VELUPV = ABS(REAL(VDRIFT)) + ABS(AIMAG(VDRIFT))
C
C      The next two statements prevent division by zero.  86400 is just a large
C      number in this case = no. secs in a day.
C
         DTSMAL = 86400.
         IF(VELUPV.GT.0.01)DTSMAL = DX/VELUPV
         IF((DTSMAL+SUMDT).GT.DTPTCL)DTSMAL = DTPTCL - SUMDT
         IPASS = IPASS + 1
         IF((SUMDT+DTSMAL).GE.DTPTCL)IPASS = 9999
         SUMDT = SUMDT + DTSMAL
         CALL RANDU(IX,IY,YFL)
         IX = IY
         ANG = PI*YFL
         CALL GAUSS(IX,1.0,0.0,VRAND)
         IF (DIFFUD.LT.0.0) DDD = 2.88*CABS(VDRIFT)
         IF (DIFFUD.GE.0.0) DDD = 4*DIFFUD
         VRAND = VRAND*SQRT(DDD/DTSMAL)
         VX = VRAND*COS(ANG)
         VY = VRAND*SIN(ANG)
         STDEV = 100.*SQRT(DDD/DTSMAL)/CABS(VDRIFT)
         VMAG = CABS(VDRIFT)
         VDRIFT = VDRIFT + CMPLX(VX,VY)
         PARTCL(I) = PARTCL(I) + DTSMAL*VDRIFT
         STDEV = 100.*VRAND/CABS(VDRIFT)
C
C     Check for spill hitting the boundaries
C
      L = REAL(PARTCL(I))/DX + 1
      M = AIMAG(PARTCL(I))/DX + 1
      IF(M.GE.IGRILB(L).AND.M.LE.IGRIUB(L))GOTO 55
      NHITB = NHITB + 1
      IHITB(NHITB) = I
55    IF(IGRLB1(L).EQ.0)GOTO 59
      IF(M.LE.IGRUB1(L).AND.M.GE.IGRLB1(L))GOTO 58
      GOTO 59
```

```
58      NHITB = NHITB+1
        IHITB(NHITB) = I
59      CONTINUE
        IF(IPASS.NE.9999)GOTO 40
60      CONTINUE
        RETURN
        END
```

Subroutine PRINT1

```
      SUBROUTINE PRINT1(IUT,NBRNCH,NGRIDX,DX)
C
C     This subroutine prints Heading and river configuration data
C     IUT defines the unit No. to which the info will be written
C     Last date of revision :  July 17, 1985
C
      character *8 DATRUN(2),TIMRUN
      COMPLEX VSTRM(99,16),CORDV(99,16),VCAR(12000),CORDLB(99)
      COMPLEX SPCEN,PARTCL(1000),VWIND,VDRIFT
      COMMON /VA/ VCAR,VWIND,VDRIFT
      COMMON /VEL/VSTRM,CORDV,CORDLB,Q(30),WL(30),TICE(99,20),
     $ YWID(99,20),Z(99,20),ZD(99),NSLSCT(99),SCTANG(99),
     $ LCSTSQ(30),NSTUBE(99),NUMCON(99),NFIRCO(99),NSECO(99),KINTM
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      WRITE(IUT,10)DATRUN,TIMRUN
      WRITE(IUT,20)NBRNCH,NGRIDX,DX,KINTM
      IS2=0
      DO 100 I=1,NBRNCH
         IS1=IS2+1
         IS2 = LCSTSQ(I)
         WRITE(IUT,30)I,IS1,IS2
100      CONTINUE
      WRITE(IUT,40)
      IS2 = IS2 +1
      DO 110 I=1,IS2
         J = NSLSCT(I)+1
         IWIDTH = YWID(I,J)
         WRITE(IUT,50)I,CORDLB(I),SCTANG(I),IWIDTH,ZD(I),NSTUBE(I),
     $   NUMCON(I),NFIRCO(I)
110      CONTINUE
      WRITE(IUT,70)
      DO 120 I=1,IS2
         IN=NSLSCT(I)+1
         WRITE(IUT,80)I,(YWID(I,J),Z(I,J),J=1,IN)
120      CONTINUE
      WRITE(IUT,60)
      DO 130 I=1,NGRIDX
         KNUM=2
         IF(IGRLB1(I).NE.0)KNUM=4
         WRITE(IUT,65)I,IGRILB(I),IGRIUB(I),IGRLB1(I),IGRUB1(I),
     $   (TYPBND(K,I),K=1,KNUM)
130      CONTINUE
10    FORMAT(1H1,///2X,75('*')/2X,75('*')/'  **',71X,'**'/'  **',14X,
     $ 'SIMULATION MODEL FOR OIL SPILLS IN RIVERS',16X,'**'/'  **',71X,
     $ '**'/'  **',7X,'DEVELOPED AT - CIVIL & ENVIR. ENG. DEPT., ',
     $ 'CLARKSON UNIVERSITY',3X,'**'/'  **',7X,'SPONSERED BY - U.S. ARMY
     $ CORPS OF ENGINEERS, DETROIT DISTRICT',3X,'**'
     $ /'  **',71X,'**'/'  **',9X,'DATE AND TIME OF RUN : ',2A8,2X,A8
     $ ,13X,'**',/'  **',71X,'**'/2X,75('*')/2X,75('*'))
20    FORMAT(///'  GEOMETRIC PROPERTIES OF RIVER'/1H+,1X,29('_')//
     $ 5X,'NO. OF BRANCHES IN UNSTEADY FLOW MODEL =',I5/
     $ 5X,'NO. OF GRIDS IN X-DIRCTION            =',I5/
     $ 5X,'GRID SIZE IN ft.',23X,'=',F6.0/
```

```
     $ 5X,'NO. OF INTERPOLATIONS BETWN SECTIONS    =',I5//
     $ 5X,'SECTIONS IN EACH BRANCH'/1H+,4X,23('_')//
     $ 5X,'BRANCH SECTIONS INVOLVED'/15X,'FROM      TO')
30    FORMAT(3(7X,I2))
40    FORMAT(1H1,//11X,'INFORMATION ON RIVER SECTIONS'/1H+,10X,29('_'),
     $ //2X,'SECTION Lower bank intersection  Angle  Width  Ref datum ',
     $ ' No str  Cond.  Connect'/12X,
     $ 'X-CORD      Y-CORD       (rad)  (ft.)  for depth  tubes   No.',
     $ '     next 1st')
50    FORMAT(4X,I2,5X,F8.1,2X,F8.1,6X,F5.3,I7,F9.2,3I8)
60    FORMAT(1H1///10X,'GRID CONFIGURATION and BOUNDARY TYPES ',
     $ 'OF SCHEMATIZED RIVER'/1H+,9X  ,58('_')//
     $ '   X',15X,'Y GRID OF',17X,'REJECTION RATE PER TIME STEP'/
     $ ,' GRID',2('   Bank 1  Bank 2  Bank 3  Bank 4   '))
65    FORMAT(I4,2X,4(3X,I3,2X),5X,4(1X,F5.4,2X))
70    FORMAT(///,10X,'Geometry of X-Sections'/1H+,9X,22('_')//
     $ ' SCTN',10X,'Distance and Depth (ft.) in pairs of data')
80    FORMAT(/I3,1X,9(F5.0,':',F5.1,3X)/4X,9(F5.0,':',F5.1,3X))
      RETURN
      END
```

```
      SUBROUTINE SPRDAX(DX,SPILDT,TIMET,INDPRN,SPAREA,SPLTIM,SPLRAT)
C
C     This subroutine handles the Axi-symmetrical spreading of spill
C     due to gravity, viscous and surface tension forces
C     This version includes spreading under ice cover.
C
C     Last date of revision : October 18, 1985
C
      COMPLEX SPCEN,PARTCL(1000)
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      COMMON /BLOCK7/SPGOIL,ANIU,SIGMA,AK2I,AK2V,AK2T,
     $ VOLPAR,VOLPIE(8),SLICKR(8)
      COMMON /SO/IMOVIN(1000),YSHIFT(1000),NMOVIN,SSHIFT
      COMMON /SE/FEVP1,FEVP2,CEVP,T0EVP
      COMMON /ICE/NICEX1(20),NICEY1(20),NICEX2(20),NICEY2(20),NICERG,
     $ AMIUO,ANICE,IPOS1(20),IPOS2(20),SPAICE
      COMMON /SPREAD/ RADIUS(1000),NTRACK(1000)
C
C     The spill area is divided into 8 pie segments around the spill
C     center.  Particles in each pie sreads according to modified
C     Fay's law's for axi-symmetrical case.
C     (see text for details)
C     Input : .. Spill center
C             .. Location of each particle
C             .. oil properties
C     Output: .. New loaction of each particle
C
C     Explanation of variables used only in this subroutine
C     VOLPIE(I)  .. an array containing the volume of oil in each pie
C                segment at previous time step NOTE: volume stored is
C                8 times the volume in pie
C     RADIUS(I) -- distance to particles in pie from spillcenter. It is
C           assumed that no more than 500 particle are in a pie any time
C     SPAREA  - Free surface area of spill (sq. ft)
C     SPAICE  - Area of spill under ice (sq. ft)
C     ICOND = 0 - Oil in the pie has free surface conditions
C           = 1 - Oil in the pie is under ice
C
      DATA PI,ROWAT,G/3.141592, 1.92, 32.2/
C
C     Evaluate some constants to be used in subsequent computations
C
      DELTA =  1.0 - SPGOIL
      AKINER = 0.25*AK2I*(DELTA*G)**0.25
      AKVISC = AK2V*(DELTA*G/SQRT(ANIU))**0.166
      AKSURF = 0.75*AK2T*(SIGMA**2/(ROWAT**2*ANIU))**0.25
      AKICE  = 0.0056666*((1-SPGOIL)*32.2*SPLRAT**2)**0.16666/ANICE
C
C     Compute the mean radius for all moving particles
C
      TOTRAD=0.
      DO 7 I =1,NMOVIN
        J=IMOVIN(I)
```

```
        TOTRAD = TOTRAD+CABS(PARTCL(J)-SPCEN)
7       CONTINUE
      TOTRAD = TOTRAD/NMOVIN
      SPX = REAL(SPCEN)
      SPY = AIMAG(SPCEN)
      SPAREA = 0.0
       SPAICE = 0.0
C
C     Loop 500 is working for one pie at a time
C
      DO 500 IPIE=1,8
      ANG1 = (IPIE-1)*PI/4.
      ANG2 = ANG1 + PI/4.
      NPTPIE = 0
      NPTICE = 0
      ICOND  = 0
      DO 10 I=1,NPTCL
10    NTRACK(I)=0
C
C     The next loop 20 is for finding the ID no's of particles belonging
C     to the pie. Radial dist. to particle from center is also computed
C     and stored in RADIUS(). NTRACK() stores the ID's of particles.
C
      DO 20 I=1,NMOVIN
        J=IMOVIN(I)
        ATX2 = REAL(PARTCL(J))-SPX
        ATX1 = AIMAG(PARTCL(J))-SPY
        ANG=ATAN2(ATX1,ATX2)
        IF(ANG.LT.0.0)ANG = ANG + 2.*PI
        IF(ANG.LT.ANG1.OR.ANG.GE.ANG2)GOTO 20
        RAD = CABS(PARTCL(J)-SPCEN)
        IF(RAD.GT.2.20*TOTRAD)GOTO 20
        NPTPIE = NPTPIE+1
        RADIUS(NPTPIE) = RAD
        NTRACK(NPTPIE) = J
        IF(NICERG.EQ.0)GOTO 20
        L = REAL(PARTCL(J))/DX
        M =(AIMAG(PARTCL(J))+YSHIFT(J))/DX
        IPOS = 0
        IF(L.EQ.0)GOTO 117
        DO 115 L1=1,L
         IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
115     CONTINUE
117     IPOS = IPOS+M-IGRILB(L+1)+3
        DO 120 K=1,NICERG
          IF(IPOS.GE.IPOS1(K).AND.IPOS.LE.IPOS2(K))NPTICE=NPTICE+1
120       CONTINUE
20      CONTINUE
C
C NO PARTICLES- NO SPREADING
C
      IF(NPTPIE.LT.1)GOTO 500
      RMEAN=0.
      DO 40 I=1,NPTPIE
40      RMEAN=RMEAN+RADIUS(I)
```

```
      RMEAN =RMEAN/NPTPIE
C
C     Check if this pie should spread as free-surface or ice conditions
C     and if it is Ice conditions is the spilling still continuing.
C
      IF(FLOAT(NPTICE)/FLOAT(NPTPIE).GT.0.5)ICOND=1
      IF(ICOND.EQ.1.AND.TIMET.GT.SPLTIM)GOTO 170
C
C     Determine the rate of spread at pie radius
C
      VOLNOW = VOLPAR*NPTPIE*8
      TIMBAR = TIMET - SPILDT/2.0
      VOLBAR =(VOLNOW+VOLPIE(IPIE))/2.0
      IF(ICOND.EQ.1)GOTO 47
      TVISC=(AK2V/AK2I)**4*(VOLBAR/(DELTA*G*ANIU))**0.333
      TERMIN=823.5*(ROWAT/SIGMA)**0.6666*SQRT(VOLBAR)*ANIU**0.3333
     $ /AK2T**1.3333
      IF(TIMBAR.GT.TERMIN)GOTO 500
      TSURFT = (AK2V/AK2T)**2*(DELTA*G*ANIU)**0.3333
     $ *(ROWAT/SIGMA)*VOLBAR**0.6666
      IF(TIMBAR.GT.TSURFT) GOTO 45
      DVDT = VOLNOW*(FEVP1-FEVP2)/SPILDT
      IF(TIMBAR.LE.TVISC) DRDT =
     $ AKINER*(DVDT+2*VOLBAR/TIMBAR)*SQRT(TIMBAR)/(VOLBAR**0.75)
      IF(TIMBAR.GT.TVISC)DRDT =
     $ AKVISC*(DVDT/3.+VOLBAR/(TIMBAR*4))*TIMBAR**0.25/VOLBAR**0.666
      GOTO 48
45    DRDT = AKSURF/(TIMBAR**0.25)
47    IF(ICOND.EQ.1)DRDT = AKICE/(TIMBAR**0.33333)
48    VOLPIE(IPIE) = VOLNOW
      SPRATE = DRDT*SPILDT/RMEAN
C
C      Rate of spreading at mean pie radius has been computed. Now spread
C      the particles in the pie proportionately.
C
      DO 140 I=1,NPTPIE
         J=NTRACK(I)
         RADOLD = CABS(PARTCL(J)-SPCEN)
         RADNEW = RADOLD*(SPRATE+1)
         IF(RADNEW.LT.0.0)RADNEW = 0.
         RADIUS(I) = RADNEW
         X = REAL(PARTCL(J)-SPCEN)
         Y = AIMAG(PARTCL(J)-SPCEN)
         X = X*RADNEW/RADOLD
         Y = Y*RADNEW/RADOLD
         PARTCL(J) = SPCEN + CMPLX(X,Y)
140      CONTINUE
      RMEAN=0.
      DO 160 I=1,NPTPIE
160      RMEAN=RMEAN+RADIUS(I)
      RMEAN =RMEAN/NPTPIE
170   SLICKR(IPIE) = RMEAN
      IF(ICOND.EQ.0)SPAREA = SPAREA + PI*RMEAN**2/8.
      IF(ICOND.EQ.1)SPAICE = SPAICE + PI*RMEAN**2/8.
      IF(INDPRN.EQ.1)WRITE(*,220)IPIE,NPTPIE,RMEAN
```

```
500     CONTINUE
C
C       Check for spill hitting the boundaries
C
        DO 60 I=1,NMOVIN
          J=IMOVIN(I)
          IF(YSHIFT(J).LT.DX)GOTO 54
          PARTCL(J)=PARTCL(J)+CMPLX(0.,YSHIFT(J))
          L=REAL(PARTCL(J))/DX + 1
          M=AIMAG(PARTCL(J))/DX +1
          IF(M.GT.IGRUB1(L))GOTO 54
          X=REAL(PARTCL(J))
          Y=IGRUB1(L)*DX-0.25*DX
          PARTCL(J)=CMPLX(X,Y)
          NHITB=NHITB+1
          IHITB(NHITB)=J
          GOTO 60
54        L = REAL(PARTCL(J))/DX + 1
          M = AIMAG(PARTCL(J))/DX + 1
          IF(M.GE.IGRILB(L).AND.M.LE.IGRIUB(L))GOTO 55
          NHITB = NHITB + 1
          IHITB(NHITB) = J
55        IF(IGRLB1(L).EQ.0)GOTO 60
          IF(M.LE.IGRUB1(L).AND.M.GE.IGRLB1(L))GOTO 58
          GOTO 60
58        NHITB = NHITB+1
          IHITB(NHITB) = J
60        CONTINUE
        RETURN
210     FORMAT(' WARNING * MAY CAUSE ERRORS PARTICLES IN PIE EXCEED 500')
220     FORMAT(I3,8X,I3,10X,F7.0)
        END
```

Subroutine SPRD1X

```
      SUBROUTINE SPRD1X(DX,SPILDT,TIMET,INDPRN,SPAREA)
      COMPLEX SPCEN,PARTCL(1000)
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      COMMON /BLOCK7/SPGOIL,ANIU,SIGMA,AK2I,AK2V,AK2T,
     $ VOLPAR,VOLPIE(8),SLICKR(8)
      COMMON /BLOCK8/AKC10,AKC20,AKC30
      COMMON /SO/IMOVIN(1000),YSHIFT(1000),NMOVIN,SSHIFT
      COMMON /SE/FEVP1,FEVP2,CEVP,T0EVP
      COMMON /ICE/NICEX1(20),NICEY1(20),NICEX2(20),NICEY2(20),NICERG,
     $ AMIUO,ANICE,IPOS1(20),IPOS2(20),SPAICE
      COMMON /SPREAD/ RADIUS(1000),NTRACK(1000)
      DIMENSION SPRATE(2),NPT(2),XLE(2)
C
C     Last date of revision : april 11, 1986
C
C     This Subroutine handles one dimensional spreading in X-direction
C     The spill area is divided into strips. Particles in each strip
C     spreads according to spreading law  for one-dimensional case.
C     (see text for details)
C     Input : .. Spill center
C             .. Location of each particle
C             .. oil properties
C     Output: .. New loaction of each particle
C
C     Explanation of variables used only in this subroutine
C     RADIUS(I) -- distance to particles in a strip from stripcenter.
C          A maximum of 500 particles can be in a strip at any time
C     IMOVIN(I) -- Index in array PARTCL of moving particles
C          ex. 1,3,4,5,7,11,12,13...... etc.
C     NMOVIN     -- Number of Moving Particles
C     SPAREA  - Free surface area of spill (sq. ft)
C     SPAICE  - Area of spill under ice (sq. ft)
C     ICOND = 0 - Oil in the strip has free surface conditions
C           = 1 - Oil in the strip is under ice
C
      DATA PI,ROWAT,G/3.141592, 1.92, 32.2/
C
C     Evaluate some constants to be used in subsequent computations
C
      DELTA =  1.0 - SPGOIL
      AKINER = AKC10*(G*DELTA/DX)**0.3333
      AKVISC = AKC20*(G*DELTA)**0.25/(SQRT(DX)*ANIU**0.125)
      AKSURF = AKC30*SQRT(SIGMA/ROWAT)/(ANIU**0.25)
C
C     To minimize some later computing, determine X-grid boxes of
C     extreme particles.
C
      LMAX=0
      LMIN=1000
      DO 40 I=1,NMOVIN
        J=IMOVIN(I)
        L=AIMAG(PARTCL(J))/DX+1
        IF(L.GT.LMAX)LMAX=L
```

```
         IF(L.LT.LMIN)LMIN=L
40       CONTINUE
      SPAREA = 0.0
      SPAICE = 0.
C
C     Loop 500 : One strip at a time
C
      DO 500 ISTRIP=LMIN,LMAX
      XBAR=0.
      NPTSTR = 0
      NPTICE = 0
      ICOND  = 0
      DO 50 I=1,NMOVIN
        J=IMOVIN(I)
        L=AIMAG(PARTCL(J))/DX+1
        IF(ISTRIP.NE.L)GOTO 50
        NPTSTR = NPTSTR+1
        NTRACK(NPTSTR) = J
        XBAR = XBAR+REAL(PARTCL(J))
C
        IF(NICERG.EQ.0)GOTO 50
        L = REAL(PARTCL(J))/DX
        M =(AIMAG(PARTCL(J))+YSHIFT(J))/DX
        IPOS = 0
        IF(L.EQ.0)GOTO 117
        DO 115 L1=1,L
         IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
115     CONTINUE
117     IPOS = IPOS+M-IGRILB(L+1)+3
        DO 120 K=1,NICERG
          IF(IPOS.GE.IPOS1(K).AND.IPOS.LE.IPOS2(K))NPTICE=NPTICE+1
120       CONTINUE
50      CONTINUE
C
C     Must have at least two particles in the strip for spreading
C
      IF(NPTSTR.LT.2)GOTO 500
      XBAR=XBAR/NPTSTR
      DO 60 I=1,NPTSTR
        J=NTRACK(I)
        RADIUS(I)= REAL(PARTCL(J))-XBAR
60      CONTINUE
      IF(FLOAT(NPTICE)/FLOAT(NPTSTR).GT.0.5)ICOND=1
C
C     XLE are the distances to the spreading edge of slick in the strip
C     computed based on the mean distance to particles from strip center.
C     index=1 for + dir from XBAR and 2 for - dir from XBAR
C
      XLE(1)=0.
      XLE(2)=0.
      NPT(2)=0
      DO 80 I=1,NPTSTR
        IF(RADIUS(I).GE.0.0)XLE(1)=XLE(1)+RADIUS(I)
        IF(RADIUS(I).GE.0.0)GOTO 80
        XLE(2)=XLE(2)+RADIUS(I)
```

```
        NPT(2)=NPT(2)+1
80      CONTINUE
      NPT(1) =NPTSTR-NPT(2)
      XLE(1) =XLE(1)/NPT(1)
      XLE(2) =XLE(2)/NPT(2)
      IF(ICOND.EQ.1)GOTO 170
C
C     If slick thickness(STHICK) is less than ultimate thickness
C     for spreading(UTHICK) then no spreading
C     NOTE that XLE(2) is always (-)ve
C
      STHICK = VOLPAR*(NPT(1)+NPT(2))/(DX*(XLE(1)-XLE(2)))
      UTHICK = 1.3458E-5 * (VOLPAR*NMOVIN)**0.25
      IF(STHICK.LT.UTHICK)GOTO 500
C
C     determine the rate of spread at mean radius(leading edge)
C
      DO 130 K=1,2
      VOLNOW = VOLPAR*NPT(K)
      TIMBAR = TIMET - SPILDT/2.0
C
C     for this first development stage use VOLBAR=VOLNOW, and DVDT=0.
      VOLBAR=VOLNOW
      DVDT = VOLNOW*(FEVP1-FEVP2)/SPILDT
C
C     TVISC -- Time in secs for transition from Inertia to Viscous
C     TSURFT-- Time in secs for transition from Viscous to Surf Tension
C     TERMIN -- Time in secs for spreading termiation
C     DRDT-- Spreading rate at leading edge (ft/sec)
C
      TSURFT = (AKVISC/AKSURF)**2.6666*VOLBAR**1.3333
      IF(TIMBAR.GT.TSURFT) GOTO 45
      TVISC=(AKVISC/AKINER)**3.4285*VOLBAR**0.5714
      IF(TIMBAR.LE.TVISC) DRDT =
     $ 0.3333*AKINER*(2+DVDT*TIMBAR/VOLBAR)*(VOLBAR/TIMBAR)**0.3333
      IF(TIMBAR.GT.TVISC)DRDT =
     $ AKVISC*(0.375+0.5*TIMBAR*DVDT/VOLBAR)*SQRT(VOLBAR)/TIMBAR**0.625
      GOTO 48
45    DRDT = 0.75*AKSURF/(TIMBAR**0.25)
48    SPRATE(K) = DRDT*SPILDT/ABS(XLE(K))
      IF(SPRATE(K).LT.-1.0)SPRATE(K)=-1.0
130   CONTINUE
C
C     Spreading rates for mean leading edges on either side has been
C     computed. Now spread the particles proportiontely.
C
      DO 140 I=1,NPTSTR
        J=NTRACK(I)
        IF(RADIUS(I).GE.0.0)XNEW=RADIUS(I)*(SPRATE(1)+1)
        IF(RADIUS(I).LT.0.0)XNEW=RADIUS(I)*(SPRATE(2)+1)
        RADIUS(I) = XNEW
        Y = AIMAG(PARTCL(J))
        PARTCL(J) = CMPLX(XBAR+XNEW,Y)
140     CONTINUE
C
```

```
C     Compute the mean distances to leading edges after spreading.
C
      XLE(1)=0.
      XLE(2)=0.
      DO 160 I=1,NPTSTR
        IF(RADIUS(I).GE.0.0)XLE(1)=XLE(1)+RADIUS(I)
        IF(RADIUS(I).LT.0.0)XLE(2)=XLE(2)+RADIUS(I)
160     CONTINUE
      XLE(1) =XLE(1)/NPT(1)
      XLE(2) =XLE(2)/NPT(2)
      SPAREA = SPAREA + DX*(XLE(1)+ ABS(XLE(2)))
170   IF(INDPRN.EQ.1)WRITE(*,220)ISTRIP,NPTSTR,XLE(2),XBAR,XLE(1)
      IF(ICOND.EQ.1) SPAICE = SPAICE + DX*(XLE(1)+ ABS(XLE(2)))
      IF(ICOND.EQ.1)WRITE(*,230)
500   CONTINUE
C
C    Move back the particles in the Northern channel which were
C    shifted by ORIENT routine.  Also check for the particles hitting
C    the boundaries
C
      DO 460 I=1,NMOVIN
        J=IMOVIN(I)
        IF(YSHIFT(J).LT.DX)GOTO 54
        PARTCL(J)=PARTCL(J)+CMPLX(0.,YSHIFT(J))
        L=REAL(PARTCL(J))/DX + 1
        M=AIMAG(PARTCL(J))/DX +1
C
C       Check for spill hitting the boundaries
C
        IF(M.GT.IGRUB1(L))GOTO 54
        NHITB=NHITB+1
        IHITB(NHITB)=J
        X=REAL(PARTCL(J))
        Y=IGRUB1(L)*DX-0.25*DX
        PARTCL(J)=CMPLX(X,Y)
        GOTO 460
54      L = REAL(PARTCL(J))/DX + 1
        M = AIMAG(PARTCL(J))/DX + 1
        IF(M.GE.IGRILB(L).AND.M.LE.IGRIUB(L))GOTO 55
        NHITB = NHITB + 1
        IHITB(NHITB) = J
        GOTO 460
55      IF(IGRLB1(L).EQ.0)GOTO 460
        IF(M.LE.IGRUB1(L).AND.M.GE.IGRLB1(L))GOTO 58
        GOTO 460
58      NHITB = NHITB+1
        IHITB(NHITB) = J
460     CONTINUE
      RETURN
220   FORMAT(I4,7X,I3,5X,F6.0,F9.0,F8.0)
230   FORMAT(1H+,50X,' ICE')
      END
```

```
      SUBROUTINE SPRD1Y(DX,SPILDT,TIMET,INDPRN,SPAREA)
      COMPLEX SPCEN,PARTCL(1000)
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /ASB/SPCEN,PARTCL,NPTCL,NHITB,IHITB(1000),TYPBND(4,300)
      COMMON /BLOCK7/SPGOIL,ANIU,SIGMA,AK2I,AK2V,AK2T,
     $ VOLPAR,VOLPIE(8),SLICKR(8)
      COMMON /BLOCK8/AKC10,AKC20,AKC30
      COMMON /SO/IMOVIN(1000),YSHIFT(1000),NMOVIN,SSHIFT
      COMMON /SE/FEVP1,FEVP2,CEVP,T0EVP
      COMMON /ICE/NICEX1(20),NICEY1(20),NICEX2(20),NICEY2(20),NICERG,
     $ AMUCO,ANICE,IPOS1(20),IPOS2(20),SPAICE
      COMMON /SPREAD/ RADIUS(1000),NTRACK(1000)
      DIMENSION SPRATE(2),NPT(2),XLE(2)
C
C     LAST DATE OF REVISION : April 11, 1986
C
C     This Subroutine handles one dimensional spreading in Y-direction
C     The spill area is divided into strips. Particles in each strip
C     spreads according to spreading law  for one-dimensional case.
C     (see text for details)
C     Input : .. Spill center
C             .. Location of each particle
C             .. oil properties
C     Output: .. New loaction of each particle
C
C     Explanation of variables used only in this subroutine
C     RADIUS(I) -  distance to particles in a strip from stripcenter.
C          A maximum of 500 particles can be in a strip at any time
C     IMOVIN(I) -- Index in array PARTCL of moving particles
C          ex. 1,3,4,5,7,11,12,13...... etc.
C     NMOVIN      -  Number of Moving Particles
C     SPAREA      Free surface area of spill (sq. ft)
C     SPAICE      Area of spill under ice (sq. ft)
C     ICOND = 0   Oil in the strip has free surface conditions
C           = 1   Oil in the strip is under ice
C
      DATA PI,ROWAT,G/3.141592, 1.92, 32.2/
C
C     Evaluate some constants to be used in subsequent computations
C
      DELTA  =  1.0 - SPGOIL
      AKINER = AKC10*(G*DELTA/DX)**0.3333
      AKVIS  = AKC20*(G*DELTA)**0.25/(SQRT(DX)*ANIU**0.125)
      AKSURF = AKC30*SQRT(SIGMA/ROWAT)/(ANIU**0.25)
C
C     To minimize some later computing, determine X-grid boxes of
C     extreme particles
C
      LMAX=0
      LMIN=1000
      DO 40 I=1,NMOVIN
        J=IMOVIN(I)
        L=REAL(PARTCL(J))/DX+1
        IF(L.GT.LMAX)LMAX=L
```

```
      IF(L.LT.LMIN)LMIN=L
40    CONTINUE
      SPAREA = 0.0
      SPAICE = 0.0
C
C     Loop 500 : One strip at a time
C
      DO 500 ISTRIP=LMIN,LMAX
      YBAR=0.
      NPTSTR = 0
      NPTICE = 0
      ICOND  = 0
      DO 50 I=1,NMOVIN
        J=IMOVIN(I)
        L=REAL(PARTCL(J))/DX+1
        IF(ISTRIP.NE.L)GOTO 50
        NPTSTR = NPTSTR+1
        NTRACK(NPTSTR) = J
        YBAR = YBAR+AIMAG(PARTCL(J))
        IF(NICERG.EQ.0)GOTO 50
        L = REAL(PARTCL(J))/DX
        M =(AIMAG(PARTCL(J))+YSHIFT(J))/DX
        IPOS = 0
        IF(L.EQ.0)GOTO 117
        DO 115 L1=1,L
         IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
115     CONTINUE
117     IPOS = IPOS+M-IGRILB(L+1)+3
        DO 120 K=1,NICERG
          IF(IPOS.GE.IPOS1(K).AND.IPOS.LE.IPOS2(K))NPTICE=NPTICE+1
120       CONTINUE
50      CONTINUE
C
C     Must have at least two particles in the strip for spreading
C
      IF(NPTSTR.LT.2)GOTO 500
      YBAR=YBAR/NPTSTR
      DO 60 I=1,NPTSTR
        J=NTRACK(I)
        RADIUS(I)= AIMAG(PARTCL(J))-YBAR
60      CONTINUE
      IF(FLOAT(NPTICE)/FLOAT(NPTSTR).GT.0.5)ICOND=1
C
C     XLE are the distances to the spreading edge of slick in the strip
C     computed based on the mean distance to particles from strip center.
C     index=1 for + dir from YBAR and 2 for - dir from YBAR
C
      XLE(1)=0.
      XLE(2)=0.
      NPT(2)=0
      DO 80 I=1,NPTSTR
        IF(RADIUS(I).GE.0.0)XLE(1)=XLE(1)+RADIUS(I)
        IF(RADIUS(I).GE.0.0)GOTO 80
        XLE(2)=XLE(2)+RADIUS(I)
        NPT(2)=NPT(2)+1
```

```
80        CONTINUE
       NPT(1) =NPTSTR-NPT(2)
       XLE(1) =XLE(1)/NPT(1)
       XLE(2) =XLE(2)/NPT(2)
       IF(ICOND.EQ.1)GOTO 170
C
C      If slick thickness(STHICK) is less than ultimate thickness
C      for spreading(UTHICK) then no spreading
C      NOTE that XLE(2) is always (-)ve
C
       STHICK = VOLPAR*(NPT(1)+NPT(2))/(DX*(XLE(1)-XLE(2)))
       UTHICK = 1.3458E-5 * (VOLPAR*NMOVIN)**0.25
       IF(STHICK.LT.UTHICK)GOTO 500
C
C      determine the rate of spread at mean radius(leading edge)
C
       DO 130 K=1,2
       VOLNOW = VOLPAR*NPT(K)
       TIMBAR = TIMET - SPILDT/2.0
C
C      for this first development stage use VOLBAR=VOLNOW, and DVDT=0.
       VOLBAR=VOLNOW
       DVDT = VOLNOW*(FEVP1-FEVP2)/SPILDT
C
C      TVISC -- Time in secs for transition from Inertia to Viscous
C      TSURFT-- Time in secs for transition from Viscous to Surf Tension
C      TERMIN -- Time in secs for spreading termiation
C      DRDT-- Spreading rate at leading edge (ft/sec)
C
       TSURFT = (AKVISC/AKSURF)**2.6666*VOLBAR**1.3333
       IF(TIMBAR.GT.TSURFT) GOTO 45
       TVISC=(AKVISC/AKINER)**3.4285*VOLBAR**0.5714
       IF(TIMBAR.LE.TVISC) DRDT =
      $ 0.3333*AKINER*(2+DVDT*TIMBAR/VOLBAR)*(VOLBAR/TIMBAR)**0.3333
       IF(TIMBAR.GT.TVISC)DRDT =
      $ AKVISC*(0.375+0.5*TIMBAR*DVDT/VOLBAR)*SQRT(VOLBAR)/TIMBAR**0.625
       GOTO 48
45     DRDT = 0.75*AKSURF/(TIMBAR**0.25)
48     SPRATE(K) = DRDT*SPILDT/ABS(XLE(K))
       IF(SPRATE(K).LT.-1.0)SPRATE(K)=-1.0
130    CONTINUE
C
C      Spreading rates for mean leading edges on either side has been
C      computed. Now spread the particles proportiontely.
C
       DO 140 I=1,NPTSTR
          J=NTRACK(I)
          IF(RADIUS(I).GE.0.0)YNEW=RADIUS(I)*(SPRATE(1)+1)
          IF(RADIUS(I).LT.0.0)YNEW=RADIUS(I)*(SPRATE(2)+1)
          RADIUS(I) = YNEW
          X = REAL(PARTCL(J))
          PARTCL(J) = CMPLX(X,YBAR+YNEW)
140       CONTINUE
C
C      Compute the mean distances to leading edges after spreading.
```

Subroutine SPRD1Y

```
C
      XLE(1)=0.
      XLE(2)=0.
      DO 160 I=1,NPTSTR
         IF(RADIUS(I).GE.0.0)XLE(1)=XLE(1)+RADIUS(I)
         IF(RADIUS(I).LT.0.0)XLE(2)=XLE(2)+RADIUS(I)
160      CONTINUE
      XLE(1) =XLE(1)/NPT(1)
      XLE(2) =XLE(2)/NPT(2)
      SPAREA = SPAREA + DX*(XLE(1)+ ABS(XLE(2)))
170   IF(INDPRN.EQ.1)WRITE(*,220)ISTRIP,NPTSTR,XLE(2),YBAR,XLE(1)
      IF(ICOND.EQ.1) SPAICE = SPAICE + DX*(XLE(1)+ ABS(XLE(2)))
      IF(ICOND.EQ.1)WRITE(*,230)
500   CONTINUE
C
C     Move back the particles in the Northern channel which were
C     shifted by ORIENT routine.  Also check for the particles hitting
C     the boundaries
C
      DO 460 I=1,NMOVIN
         J=IMOVIN(I)
         IF(YSHIFT(J).LT.DX)GOTO 54
         PARTCL(J)=PARTCL(J)+CMPLX(0.,YSHIFT(J))
         L=REAL(PARTCL(J))/DX + 1
         M=AIMAG(PARTCL(J))/DX +1
C
C        Check for spill hitting the boundaries
C
         IF(M.GT.IGRUB1(L))GOTO 54
         NHITB=NHITB+1
         IHITB(NHITB)=J
         X=REAL(PARTCL(J))
         Y=IGRUB1(L)*DX-0.25*DX
         PARTCL(J)=CMPLX(X,Y)
         GOTO 460
54       L = REAL(PARTCL(J))/DX + 1
         M = AIMAG(PARTCL(J))/DX + 1
         IF(M.GE.IGRILB(L).AND.M.LE.IGRIUB(L))GOTO 55
         NHITB = NHITB + 1
         IHITB(NHITB) = J
         GOTO 460
55       IF(IGRLB1(L).EQ.0)GOTO 460
         IF(M.LE.IGRUB1(L).AND.M.GE.IGRLB1(L))GOTO 58
         GOTO 460
58       NHITB = NHITB+1
         IHITB(NHITB) = J
460      CONTINUE
      RETURN
220   FORMAT(I4,7X,I3,5X,F6.0,F9.0,F8.0)
230   FORMAT(1H+,50X,' ICE')
      END
```

```
      SUBROUTINE VELDIS(IPROPT,NBRNCH,NGRIDX,DX)
C
C    This program computes the Velocity along and across the river
C    (Two-Dimensional velocity distribution)
C
C      Last Date of Revision: Jul 17, 1985
C
      COMPLEX COMPXY,VSTRM(99,16),CORDV(99,16),VCAR(12000),CORDLB(99)
      COMPLEX VWIND,VDRIFT
      COMMON /VEL/VSTRM,CORDV,CORDLB,Q(30),WL(30),TICE(99,20),
     $ YWID(99,20),Z(99,20),ZD(99),NSLSCT(99),SCTANG(99),
     $ LCSTSQ(30),NSTUBE(99),NUMCON(99),NFIRCO(99),NSECO(99),KINTM
      COMMON /VA/ VCAR,VWIND,VDRIFT
      COMMON /VASB/IGRILB(300),IGRIUB(300),IGRLB1(300),IGRUB1(300)
      COMMON /V/IZRBX(100),IZRBY(100),NZRVB
C
C    Input  : Q & WL both of which are arrays of size at least NBRNCH
C             and NBRNCH+1 respec.
C
C    Output : x & y components of velocity in the river for each grid box
C             Also computed are velocities at sections perpendicular to
C             stream thalweg and co-ordinates of the position at which
C             they are acting
C
C    This program computes velocities in the following manner
C     1 Go from branch to branch - (Branch here refers to branches in
C       Unsteady flow model
C     2 Then do for each section in a branch
C     3 Finally scans across the river, stremtube by tube
C       The above numbers also show the looping sequence where 1 is the
C       outermost and 3 is the innermost
C
      IS2=0
      DO 80 IB=1,NBRNCH
       IS1=IS2+1
       IS2=LCSTSQ(IB)
       TBRLEN=0.
       DO 30 IS=IS1,IS2
        ISCON = NFIRCO(IS)
        TBRLEN=TBRLEN+CABS(CORDLB(IS)-CORDLB(ISCON))
30      CONTINUE
       IF(IB.EQ.NBRNCH)IS2=IS2+1
       SCTLEN=0.
       DO 80 IS=IS1,IS2
        QSTUBE=Q(IB)/NSTUBE(IS)
        ATUBE1=0.
        YSTB1 = 0.
        ISCON = NFIRCO(IS)
        SCTLEN=SCTLEN+CABS(CORDLB(IS)-CORDLB(ISCON))
        IBCON = IB+1
        IF(NUMCON(IS2).NE.21)GOTO 38
        IF(NSECO(IS2).EQ.999)GOTO 38
36      IBCON = IBCON+1
        LASTSC = LCSTSQ(IBCON-1)
```

```
      IF(NUMCON(LASTSC).NE.21)GOTO 36
38    IF(IS2.EQ.NFIRCO(IS2-1).AND.(IS2-1).EQ.NFIRCO(IS2))IBCON=IB
      WLSCT=WL(IB)-(WL(IB)-WL(IBCON))*SCTLEN/TBRLEN
      SARIY=0.
      NIY=NSLSCT(IS)+1
      SXAREA = 0.
      DO 40 IY=2,NIY
c     WRITE(*,*)IY,YWID(IS,IY),YWID(IS,IY-1)
       DYRS=YWID(IS,IY)-YWID(IS,IY-1)
       PERI=SQRT(DYRS**2 + (Z(IS,IY)-Z(IS,IY-1))**2)
       ICEIND=0
       IF(TICE(IS,IY).GT.0.001.AND.TICE(IS,IY-1).GT.0.001)ICEIND=1
       TISUM = TICE(IS,IY)+TICE(IS,IY-1)
       IF(ICEIND.EQ 1)PERI=PERI+DYRS
       IF(ICEIND.EQ.0)TISUM=0.0
       AIY=DYRS*((Z(IS,IY)+Z(IS,IY-1)-TISUM)/2.+WLSCT-ZD(IS))
       IF(AIY.LT.0.0)AIY = 0.0
       HR=AIY/PERI
       SARIY=SARIY+AIY*HR**0.6666
       SXAREA = SXAREA + AIY
40     CONTINUE
      NSTUB1 = NSTUBE(IS)-1
      DO 70 ITB=1,NSTUB1
       QSET=QSTUBE*ITB
       PSARIY=0.
       SPERI =0.
       SAIY  =0.
       DO 60 IY=2,NIY
        DYRS=YWID(IS,IY)-YWID(IS,IY-1)
        PERI=SQRT(DYRS**2 + (Z(IS,IY)-Z(IS,IY-1))**2)
        ICEIND=0
        IF(TICE(IS,IY).GT.0.001.AND.TICE(IS,IY-1).GT.0.001)ICEIND=1
        TISUM = TICE(IS,IY)+TICE(IS,IY-1)
        IF(ICEIND.EQ.1)PERI=PERI+DYRS
        IF(ICEIND.EQ.0)TISUM=0.0
        AIY =DYRS*((Z(IS,IY)+Z(IS,IY-1)-TISUM)/2. + WLSCT - ZD(IS))
        IF(AIY.LT.0.0)AIY = 0.0
        HR   = AIY/PERI
        ARIY=AIY*HR**0.6666
        PSARIY= PSARIY + ARIY
        SPERI = SPERI + PERI
        SAIY  = SAIY + AIY
        QIY   = Q(IB)*PSARIY/SARIY
        IF(QIY.LT.QSET)GOTO 60
        QIY1 = Q(IB)*(PSARIY-ARIY)/SARIY
        YSTB2 = YWID(IS,IY-1)+DYRS*(QSET-QIY1)/(QIY-QIY1)
        YSTB = (YSTB1+YSTB2)/2.
        YSTB1 = YSTB2
        ATUBE = SAIY-AIY+AIY*(YSTB2-YWID(IS,IY-1))/DYRS
        VSTRM(IS,ITB) = CMPLX(QSTUBE/(ATUBE-ATUBE1),0.)
        ATUBE1 = ATUBE
        ANGL= SCTANG(IS)
        CORDV(IS,ITB)=CORDLB(IS)+CMPLX(YSTB*COS(ANGL),YSTB*SIN(ANGL))
        GOTO 65
60      CONTINUE
```

```
65       CONTINUE
70      CONTINUE
        NSTB=NSTUBE(IS)
        VSTRM(IS,NSTB)=CMPLX(QSTUBE/(SXAREA-ATUBE1),0.)
        YSTB = (YWID(IS,NIY)+YSTB1)/2.
        CORDV(IS,NSTB)=CORDLB(IS)+CMPLX(YSTB*COS(ANGL),YSTB*SIN(ANGL))
80      CONTINUE
C
C   At this point 2-D stream velocity (Along the river section by section
C     and across the river streamtube by streamtube) is assigned to
C     VSTRM's x-component. Therefore it has the correct magnitude but not
C     the direction. Later this magnitude will be correctly distributed
C     x & y components so that it has correct direction.
C   CORDV stores the location at which VSTRM is acting
C   Note : CORDV and VSTRM both are 2-D COMPLEX arrays
C
C   Now assign the correct direction to velocities
      IS2=LCSTSQ(NBRNCH)
      DO 100 IS=1,IS2
        NSTB = NSTUBE(IS)
        DO 100 ITB=1,NSTB
          NFIRST=NFIRCO(IS)
          ISCON =NFIRST
          IF(ITB.GT.NSTUBE(NFIRST))ISCON=NSECO(IS)
          ITBCON=ITB
          IF(NUMCON(IS).EQ.11)GOTO 97
          IF(NUMCON(IS).EQ.12.AND.ITB.GT.NSTUBE(NFIRST))ITBCON=
     $    ITB-NSTUBE(NFIRST)
          IF(NUMCON(IS).NE.21)GOTO 97
          IF(NSECO(IS).NE.999)GOTO 97
          DO 93 I =1,999
            J = IS -I
            IF(NSECO(J).NE.0)GOTO 95
93          CONTINUE
95        ITBCON = ITB + NSTUBE(J-1)
97        VMAG=REAL(VSTRM(IS,ITB))
          COMPXY = CORDV(ISCON,ITBCON)-CORDV(IS,ITB)
          RAD = CABS(COMPXY)
          VVX = VMAG*REAL(COMPXY)/RAD
          VVY = VMAG*AIMAG(COMPXY)/RAD
          VSTRM(IS,ITB) = CMPLX(VVX,VVY)
100     CONTINUE
C
C   The next segment writes velocities and co-ords to a file if IPROPT=1
C   This information can be used by program DIRPLOT to plot velocities
C
      IF(IPROPT.EQ.0)GOTO 415
      DO 110 IS=1,IS2
        NSTB = NSTUBE(IS)
        DO 110 ITB=1,NSTB
          WRITE(3,2100)CORDV(IS,ITB),VSTRM(IS,ITB)
110       CONTINUE
C
C   From the velocities computed at stream cross sections now assign the
C   velocities to each grid center in the Cartesian System
```

subroutine VELDIS

```
C   First assign the velocity to a grid box if co-ords are within the box
C
415   DO 120 IS =1,IS2
        NSTB = NSTUBE(IS)
        DO 120 ITB = 1,NSTB
          L = REAL(CORDV(IS,ITB))/DX
          M =AIMAG(CORDV(IS,ITB))/DX
          IPOS = 0
          IF(L.EQ.0)GOTO 117
          DO 115 L1=1,L
            IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
115         CONTINUE
117       IPOS = IPOS+M-IGRILB(L+1)+3
          VMAG = CABS(VCAR(IPOS))
          IF(VMAG.LE.0.001)VCAR(IPOS) = VSTRM(IS,ITB)
          IF(VMAG.GT.0.001)VCAR(IPOS) = (VCAR(IPOS)+VSTRM(IS,ITB))/2.
120   CONTINUE
C
C   Now check for the boxes with no assigned velocity yet;
C    For KINTM intermediate Sections interpolate in Streamtube between
C    Two adjacent X-scetions and assign a weighted mean velocity
C
      DO 130 IS=1,IS2
        NSTB = NSTUBE(IS)
        DO 130 ITB = 1,NSTB
          NFIRST=NFIRCO(IS)
          ISCON =NFIRST
          IF(ITB.GT.NSTUBE(NFIRST))ISCON=NSECO(IS)
          ITBCON=ITB
          IF(NUMCON(IS).EQ.11)GOTO 197
          IF(NUMCON(IS).EQ.12.AND.ITB.GT.NSTUBE(NFIRST))ITBCON=
     $    ITB-NSTUBE(NFIRST)
          IF(NUMCON(IS).NE.21)GOTO 197
          IF(NSECO(IS).NE.999)GOTO 197
          DO 193 I =1,999
            J = IS -I
            IF(NSECO(J).NE.0)GOTO 195
193         CONTINUE
195       ITBCON = ITB + NSTUBE(J-1)
197       CONTINUE
        DO 130 K=1,KINTM
          COMPXY= ((KINTM+1-K)*CORDV(IS,ITB)+K*CORDV(ISCON,ITBCON))
     $    /(KINTM+1)
          L = REAL(COMPXY)/DX
          M =AIMAG(COMPXY)/DX
          IPOS = 0
          IF(L.EQ.0)GOTO 127
          DO 125 L1=1,L
            IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
125         CONTINUE
127       IPOS = IPOS+M-IGRILB(L+1)+3
          VMAG = CABS(VCAR(IPOS))
          IF(VMAG.LE.0.001)VCAR(IPOS)=
     $    ((KINTM+1-K)*VSTRM(IS,ITB)+K*VSTRM(ISCON,ITBCON))/(KINTM+1)
130     CONTINUE
```

```
C
C     There may still be boxes without any assigned velocities
C     Now velocities will be assigned based on the average value of the
C     surrounding boxes
C     Start from column 2 and then move to subsequent ones. The first
C     column is neglected. Before the process begins a value of 0.0011 is
C     assigned to the grids just outside the boundary (a technique used
C     purely to simplify computations).
C
      IY1=1
      DO 133 I=1,NGRIDX
        IY2=IGRIUB(I)-IGRILB(I)+2+IY1
        VCAR(IY1)=0.0011
        VCAR(IY2)=0.0011
        IY1 = IY2+1
133     CONTINUE
      IY2 = IGRIUB(1)- IGRILB(1)+2
      DO 150 L=2,NGRIDX
        IY1 = IY2+3
        IY2 = IGRIUB(L) - IGRILB(L)+IY1
        DO 150 M = IY1,IY2
          COMPXY = (0.,0.)
          COUNT  = 0.
          IROW = M-IY1+IGRILB(L)
          IF(IGRLB1(L).EQ.0)GOTO 141
          IF(IROW.GE.IGRLB1(L).AND.IROW.LE.IGRUB1(L))VCAR(M)=0.0011
141       IF(CABS(VCAR(M)).GT.0.001)GOTO 150
C
C          IF((IGRILB(L-1)-IGRILB(L)).GT.2)GOTO 142
C          IF((IGRIUB(L)-IGRIUB(L-1)).GT.2)GOTO 142
          IF((IGRILB(L-1)-IROW).GT.2)GOTO 142
          IF((IROW-IGRIUB(L-1)).GT.2)GOTO 142
C
          MM = M+IGRILB(L)-IGRIUB(L-1) - 3
          IF(CABS(VCAR(MM)).LE.0.001)GOTO 142
          COMPXY=COMPXY+VCAR(MM)
          COUNT = COUNT+1
142       MM = M+IGRIUB(L)-IGRILB(L+1)+3
C
C          IF((IGRILB(L+1)-IGRILB(L)).GT.2)GOTO 144
C          IF((IGRIUB(L)-IGRIUB(L+1)).GT.2)GOTO 144
          IF((IGRILB(L+1)-IROW).GT.2)GOTO 144
          IF((IROW-IGRIUB(L+1)).GT.2)GOTO 144
C
          IF(CABS(VCAR(MM)).LE.0.001)GOTO 144
         COMPXY=COMPXY+VCAR(MM)
          COUNT = COUNT+1
144       IF(CABS(VCAR(M-1)).LE.0.001)GOTO 146
          COMPXY=COMPXY+VCAR(M-1)
          COUNT = COUNT+1
146       IF(CABS(VCAR(M+1)).LE.0.001)GOTO 148
          COMPXY=COMPXY+VCAR(M+1)
          COUNT = COUNT+1
148       VCAR(M)=COMPXY/COUNT
150     CONTINUE
```

subroutine VELDIS

```
C
C     For the boxes defined thru input data set VCAR=0.0
C
      DO 164 IBOX=1,NZRVB
        IF(NZRVB.GT.100)GOTO 164
        L = IZRBX(IBOX) - 1
        M = IZRBY(IBOX) - 1
        IPOS = 0
        IF(L.EQ.0)GOTO 163
        DO 160 L1=1,L
         IPOS = IPOS+IGRIUB(L1)-IGRILB(L1)+3
160     CONTINUE
163     IPOS = IPOS+M-IGRILB(L+1)+3
        VCAR(IPOS) = 0.0
164   CONTINUE
      IF(IPROPT.EQ.0)RETURN
      J1=2
      DO 170 I=1,NGRIDX
      X = I*DX - 0.5*DX
      J2 = IGRIUB(I) - IGRILB(I)+J1
        DO 165 J=J1,J2
        Y = (IGRILB(I)+J-J1)*DX-0.5*DX
        WRITE(4,2100)X,Y,VCAR(J)
165   CONTINUE
      J1=J2+3
170   CONTINUE
      RETURN
123   FORMAT(3I5,3F8.2,2F10.0)
2100  FORMAT(2F9.0,2F7.2)
      END
```

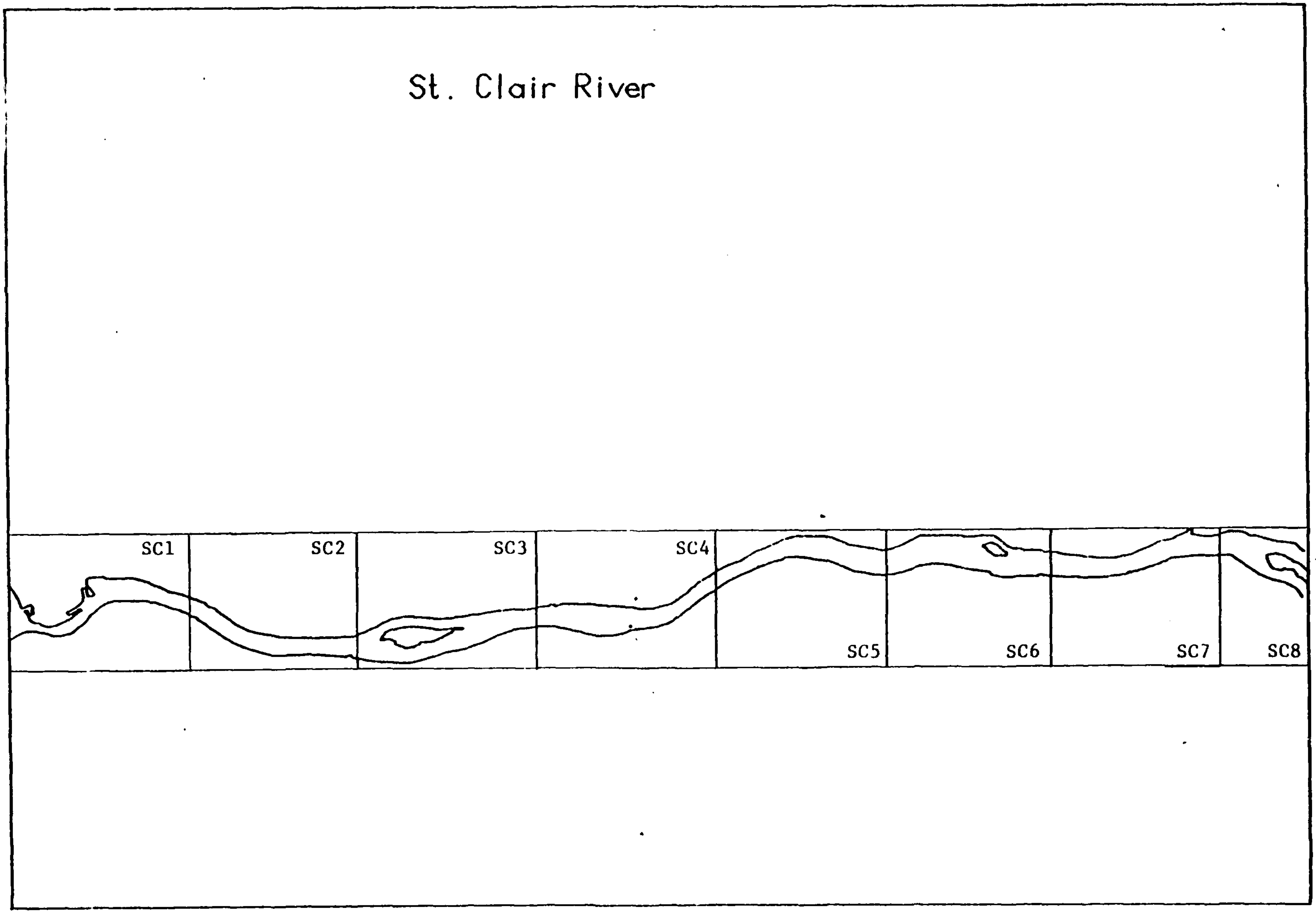
Subroutine GAUSS and RANDU

```
      SUBROUTINE GAUSS(IX,S,AM,V)
      A=0.0
      DO 50 I=1,12
      CALL RANDU(IX,IY,Y)
      IX=IY
50    A=A+Y
      V=(A-6.0)*S+AM
      RETURN
      END


      SUBROUTINE RANDU(IX,IY,YFL)
      IY = IX*65539
      IF(IY)5,6,6
5       IY = IY + 2147483647+1
6       YFL = IY
      YFL = YFL*0.4656613E-9
      YFL=RND(-1)
      RETURN
      END
```

APPENDIX III

DIGITIZED GRID SYSTEMS FOR CONNECTING CHANNELS



Index Map for Grid System in St. Clair River

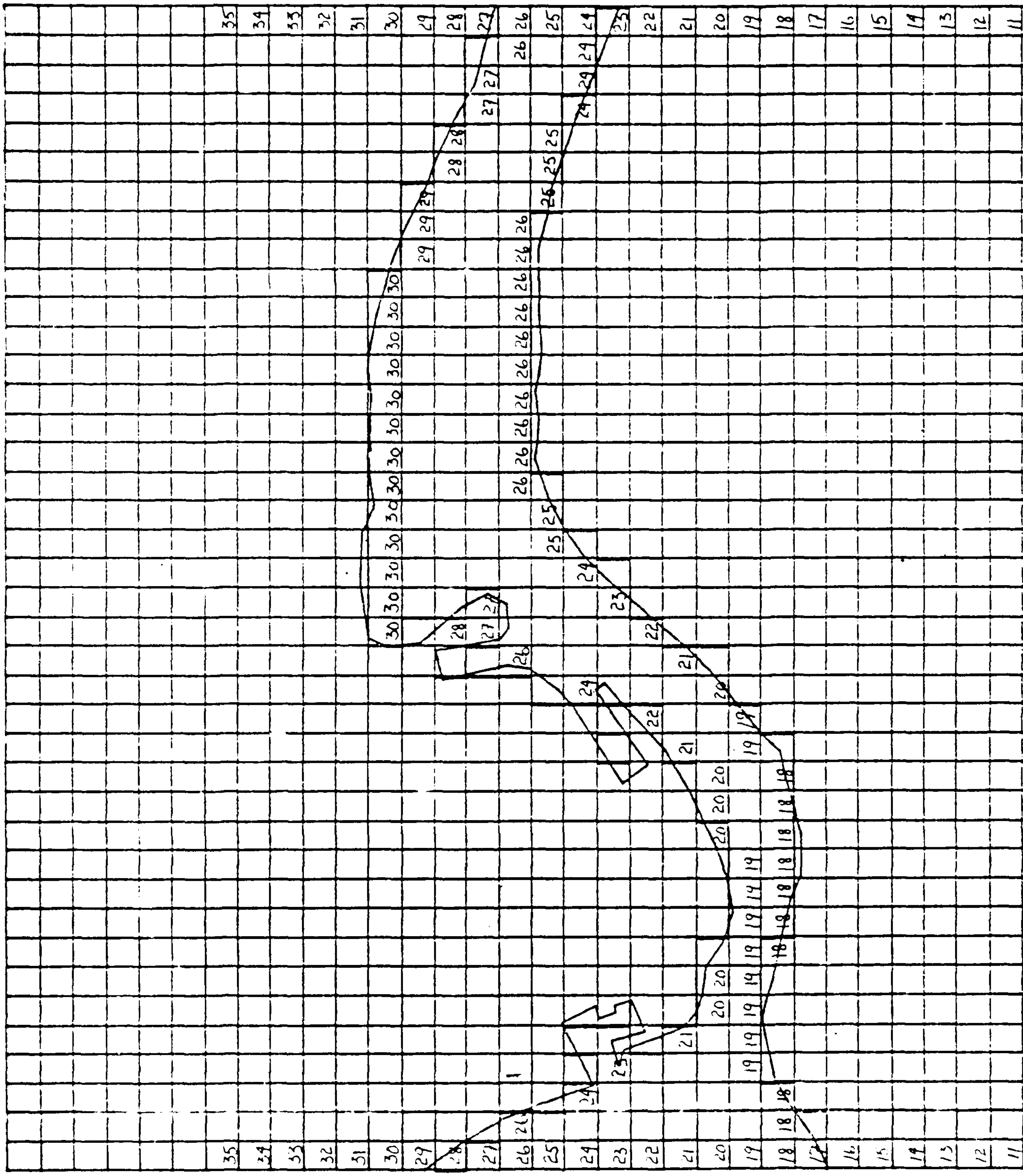
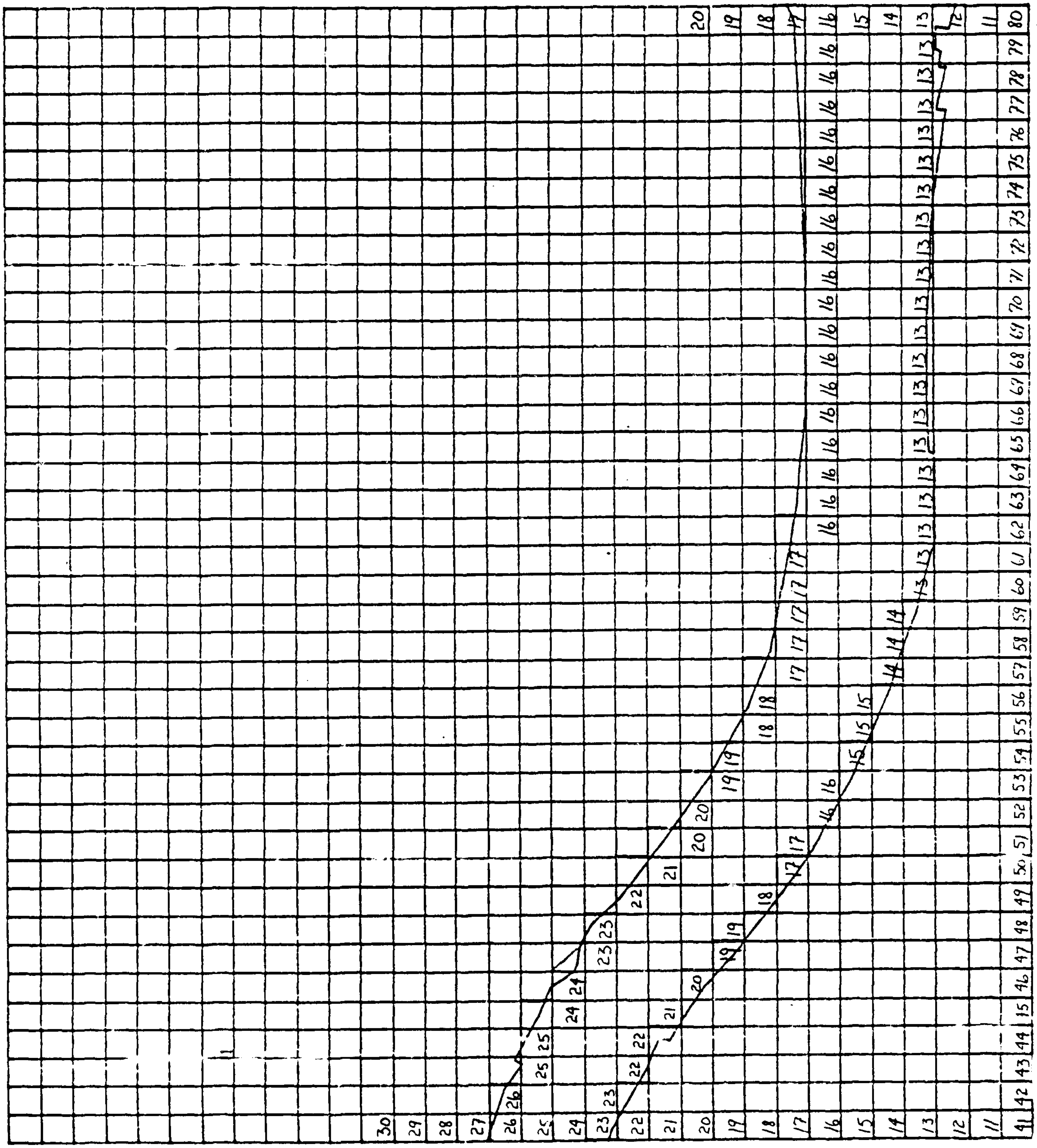


Chart SC1



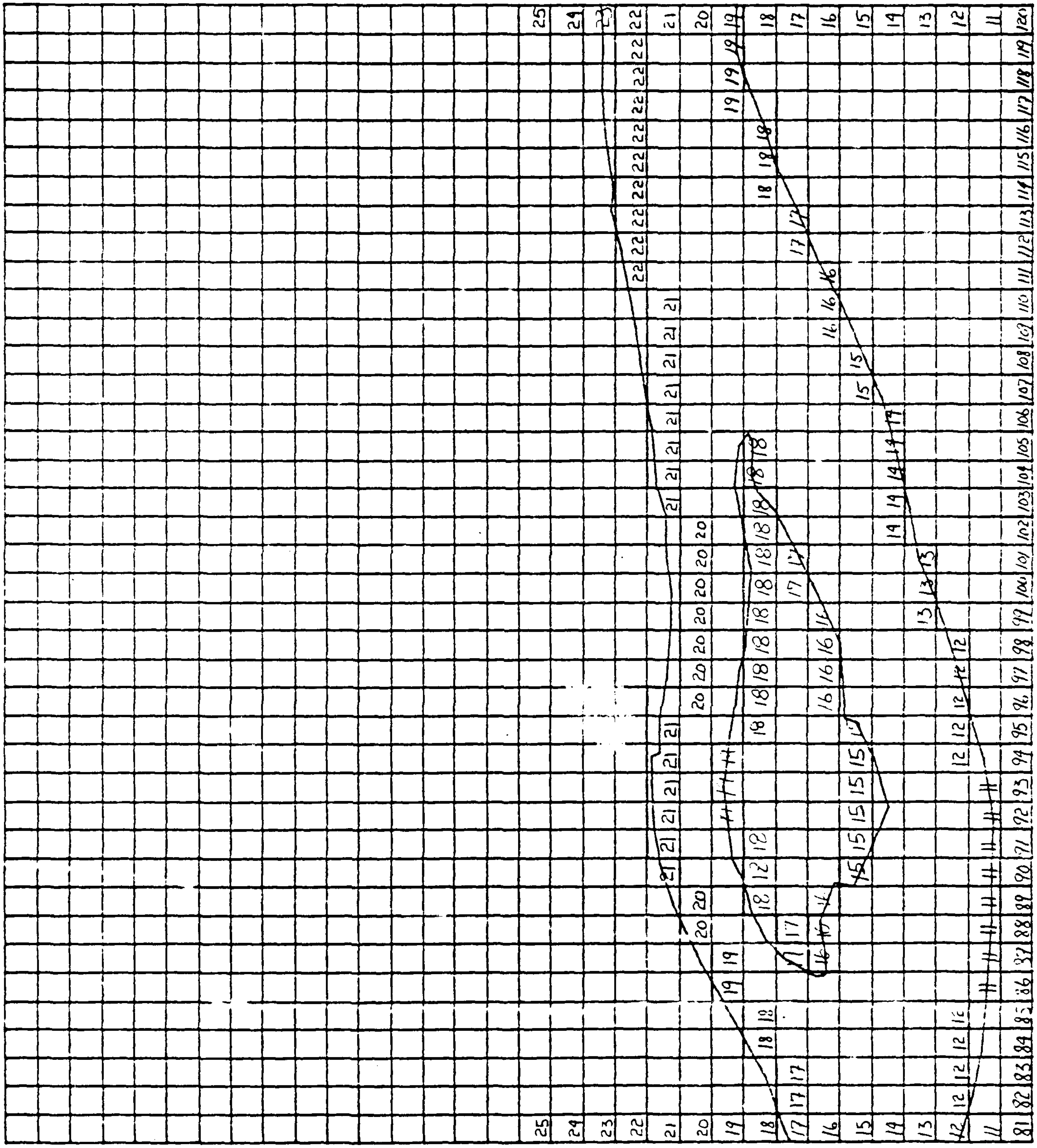
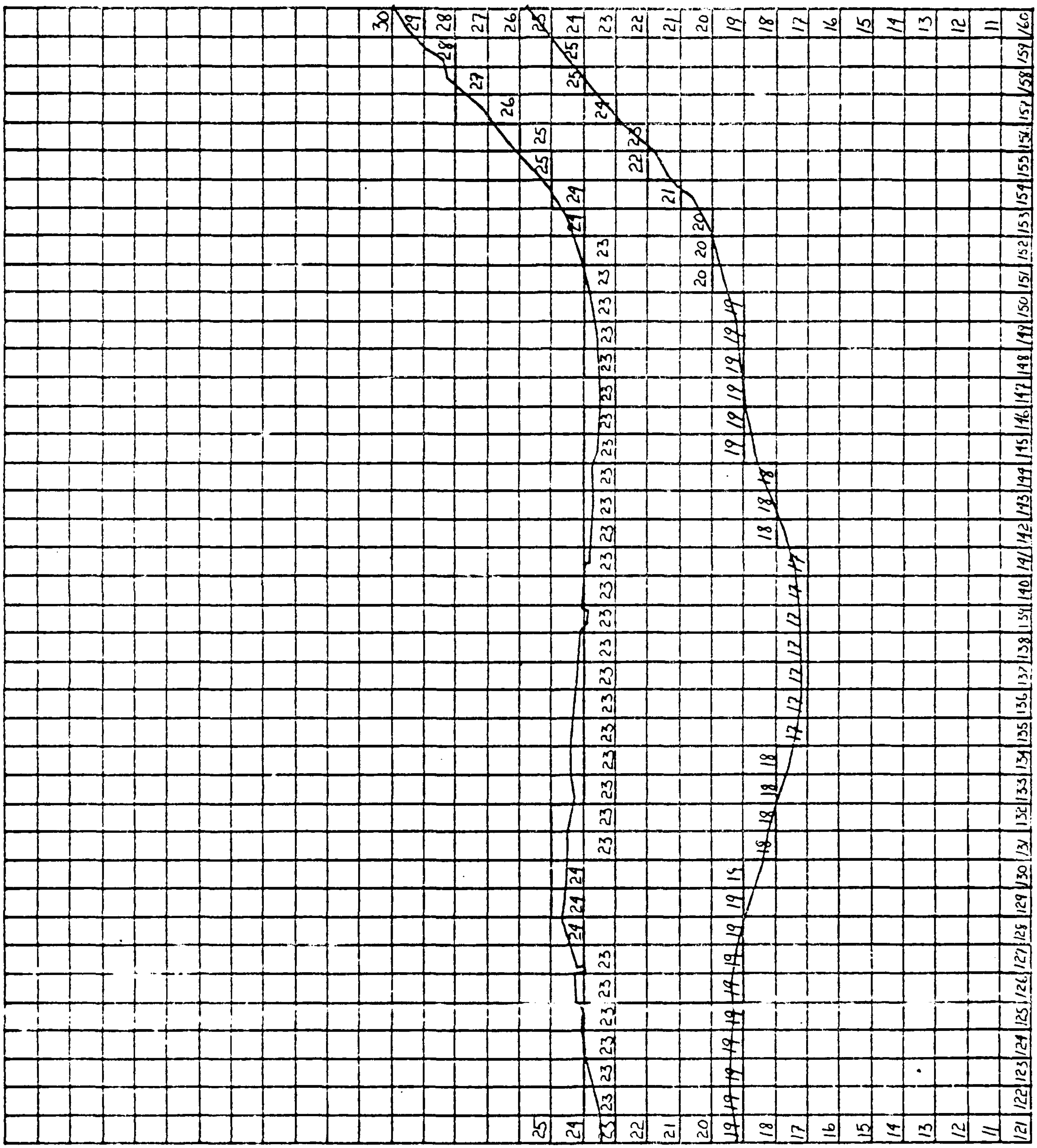


Chart SC3



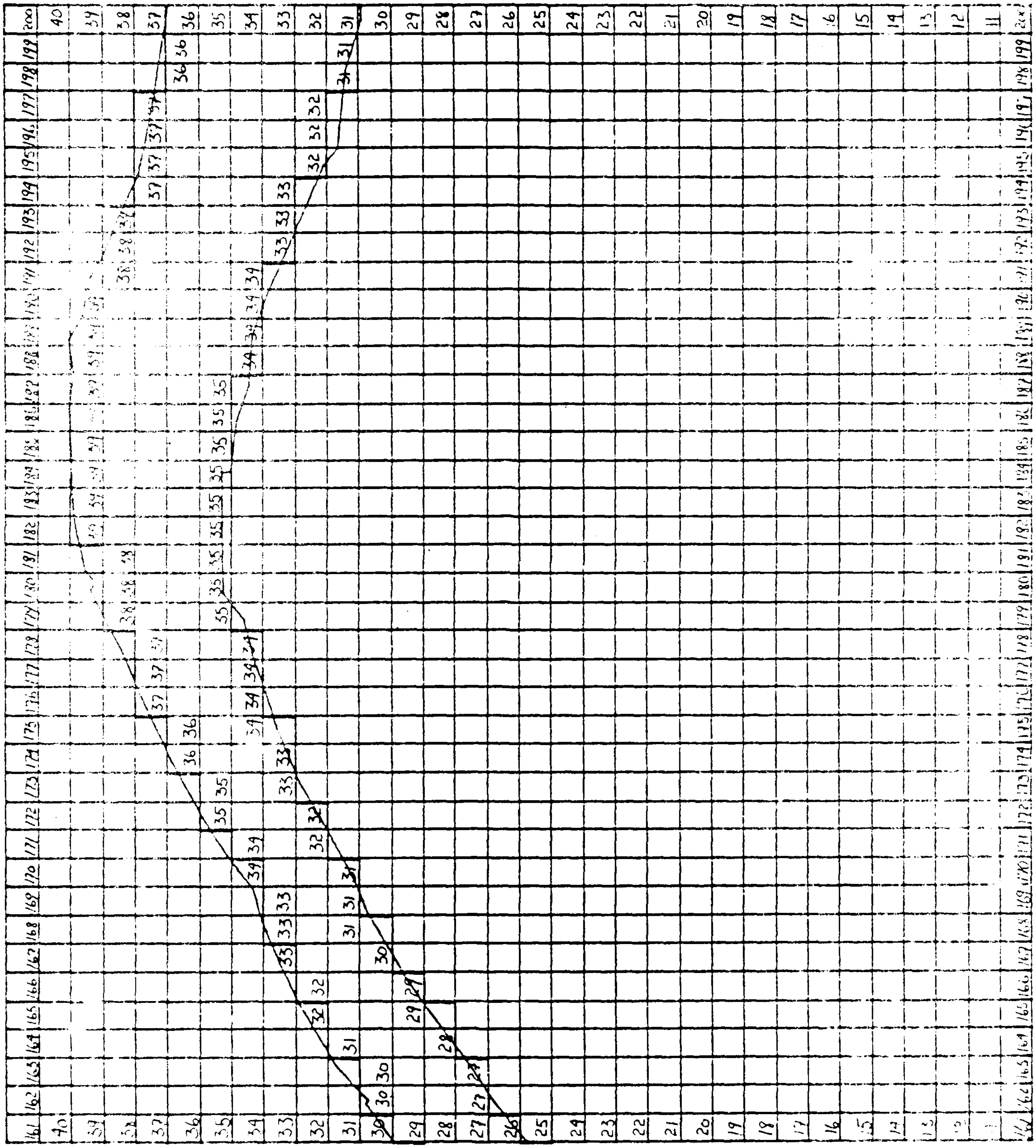


Chart SC5

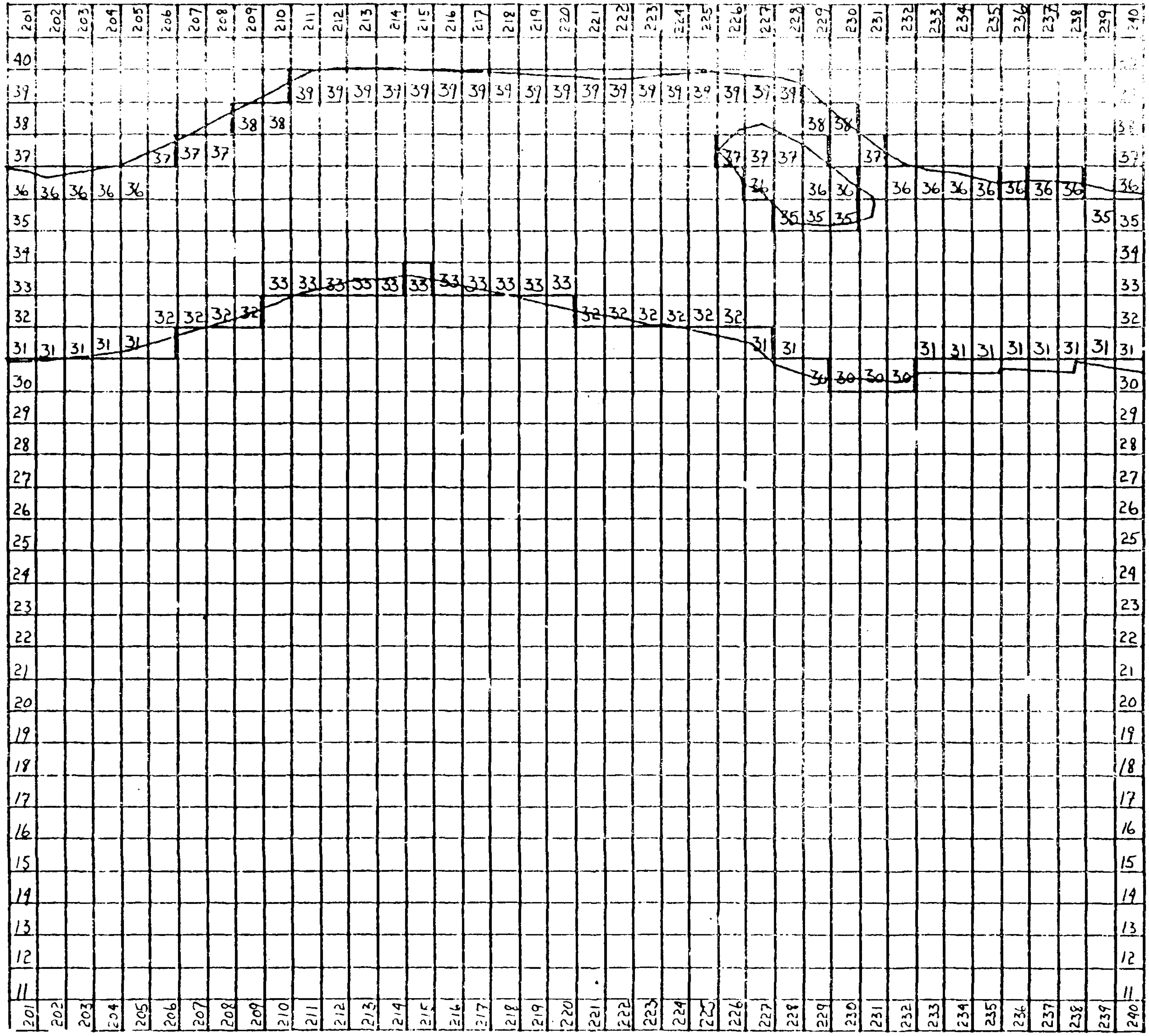


Chart SC6

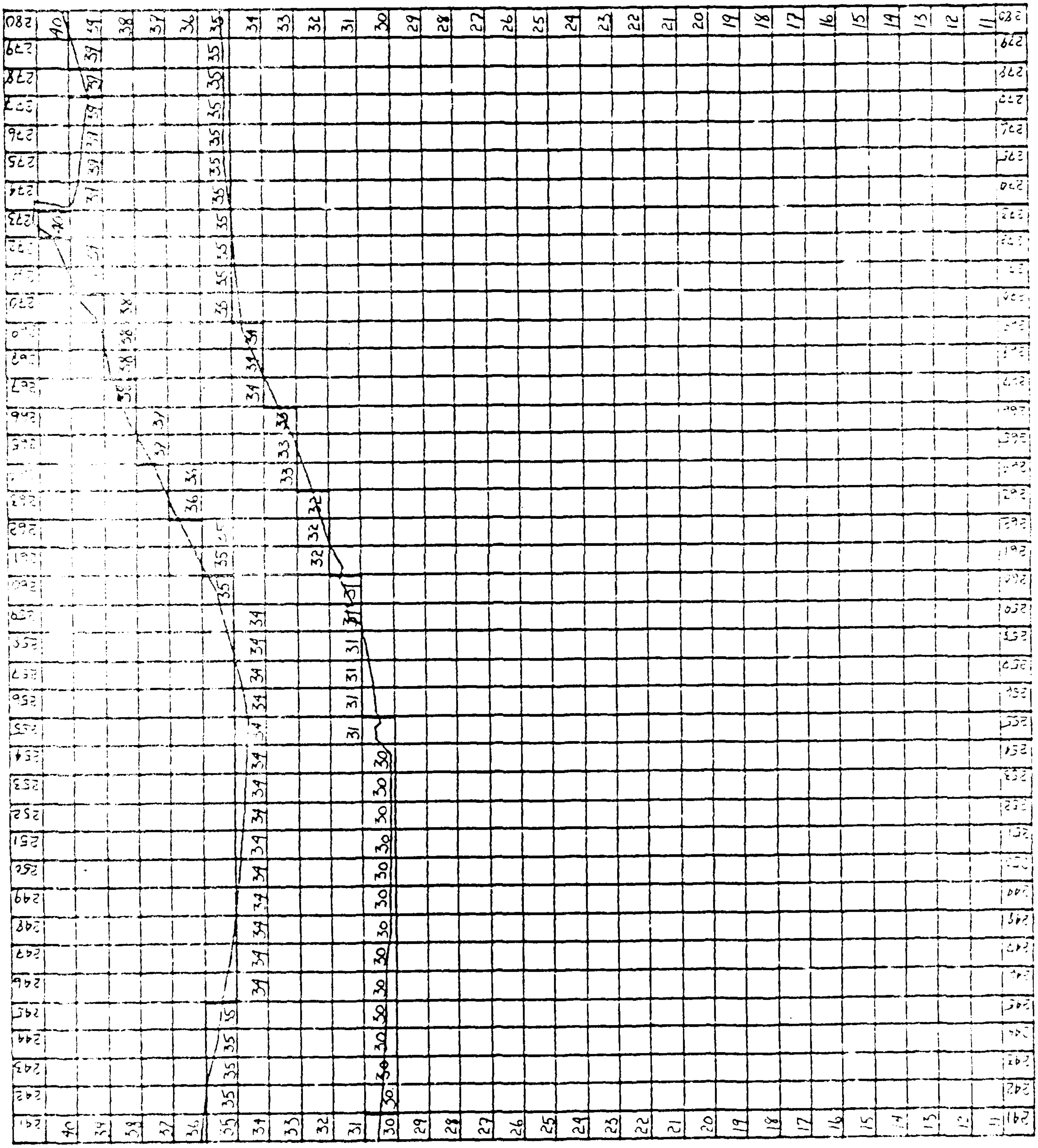
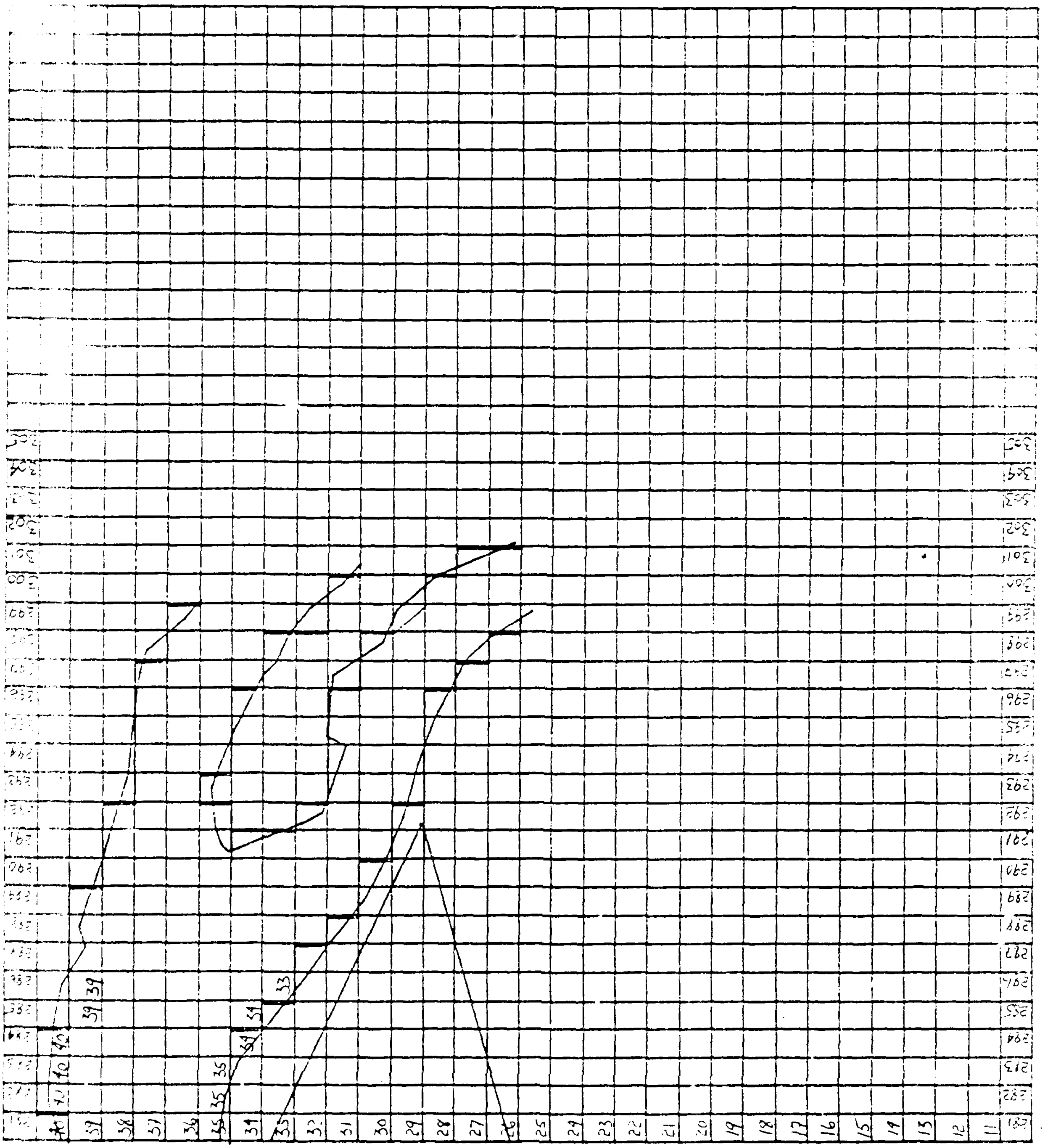
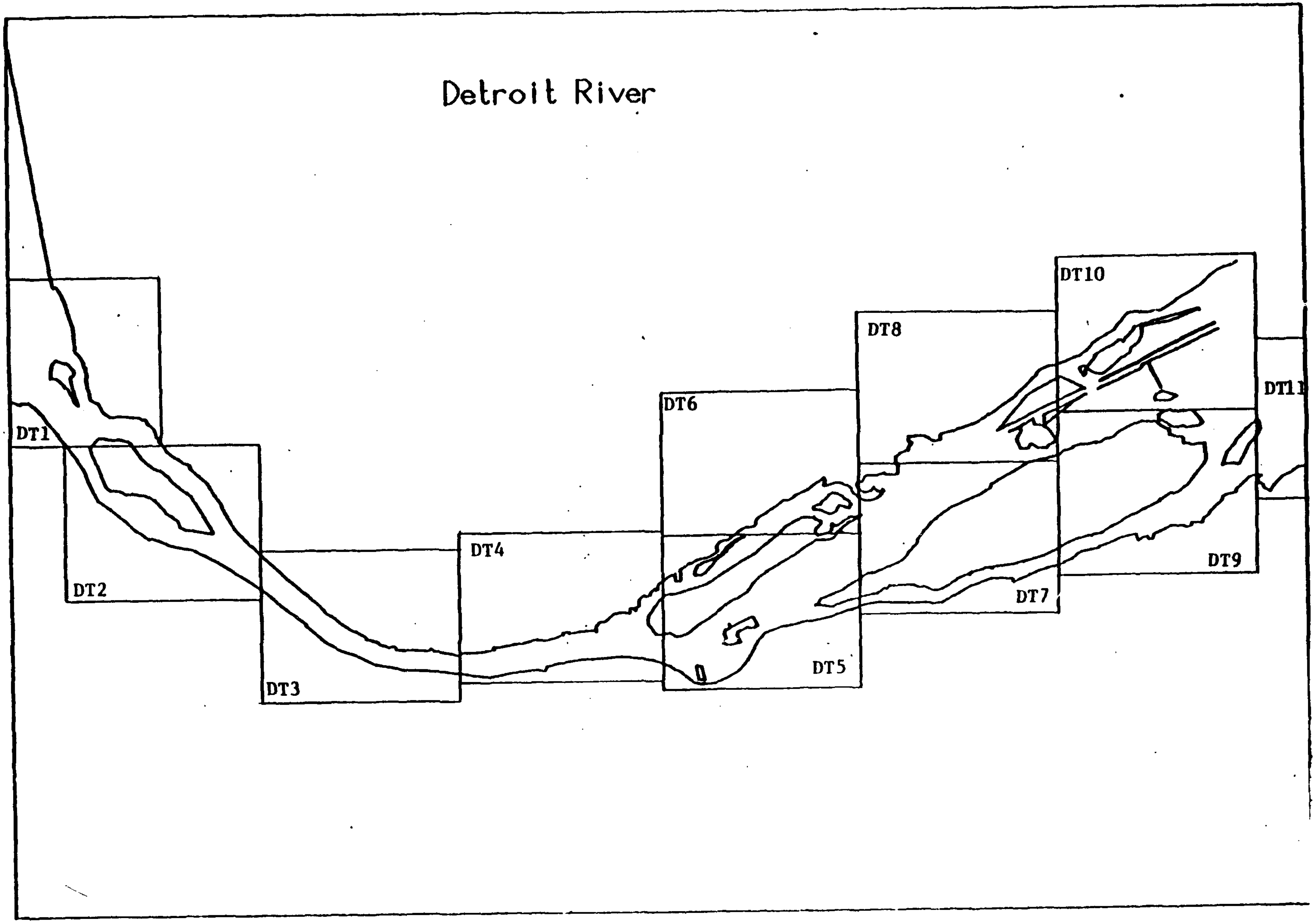
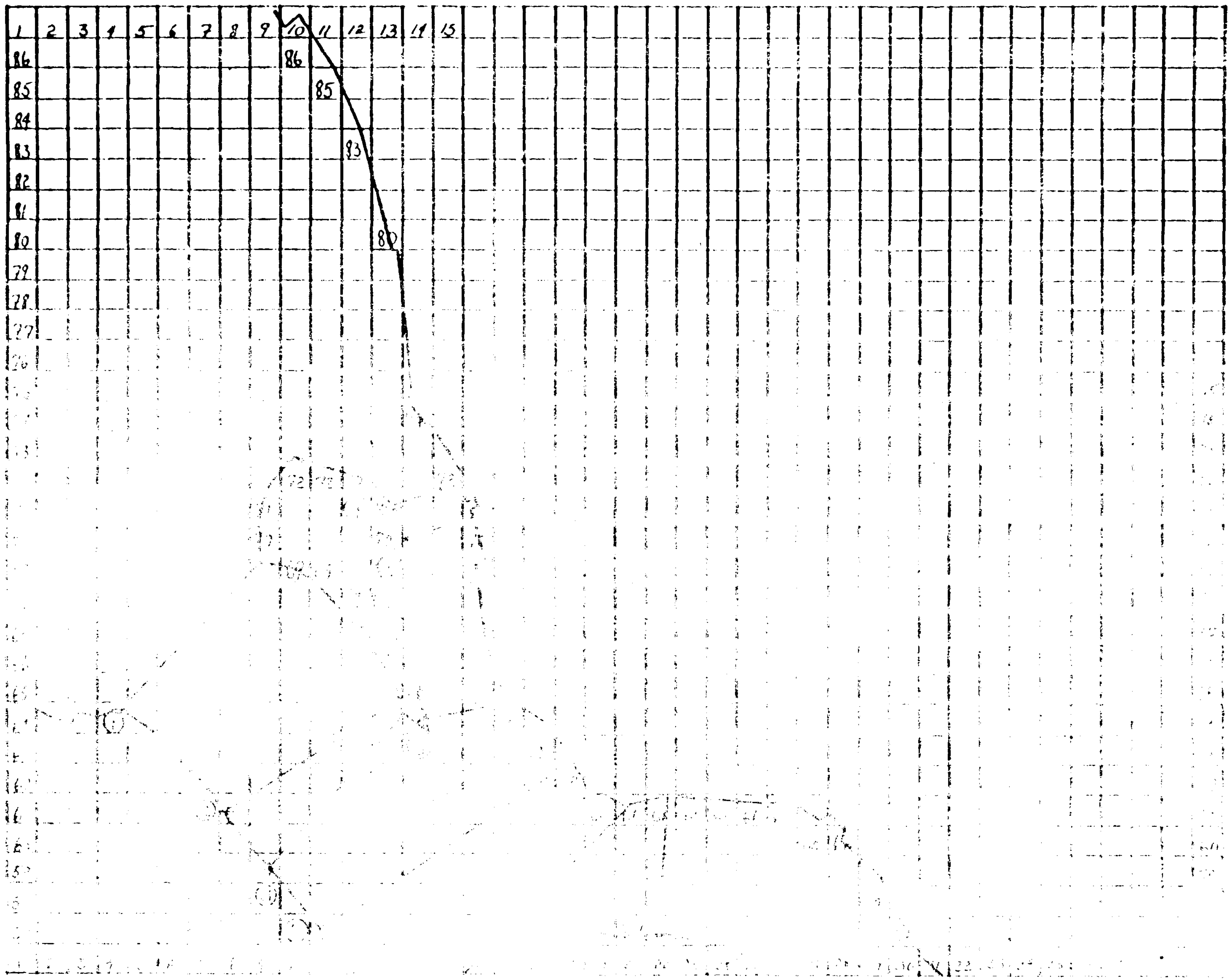


Chart SC7





Index Map for Grid System in Detroit River



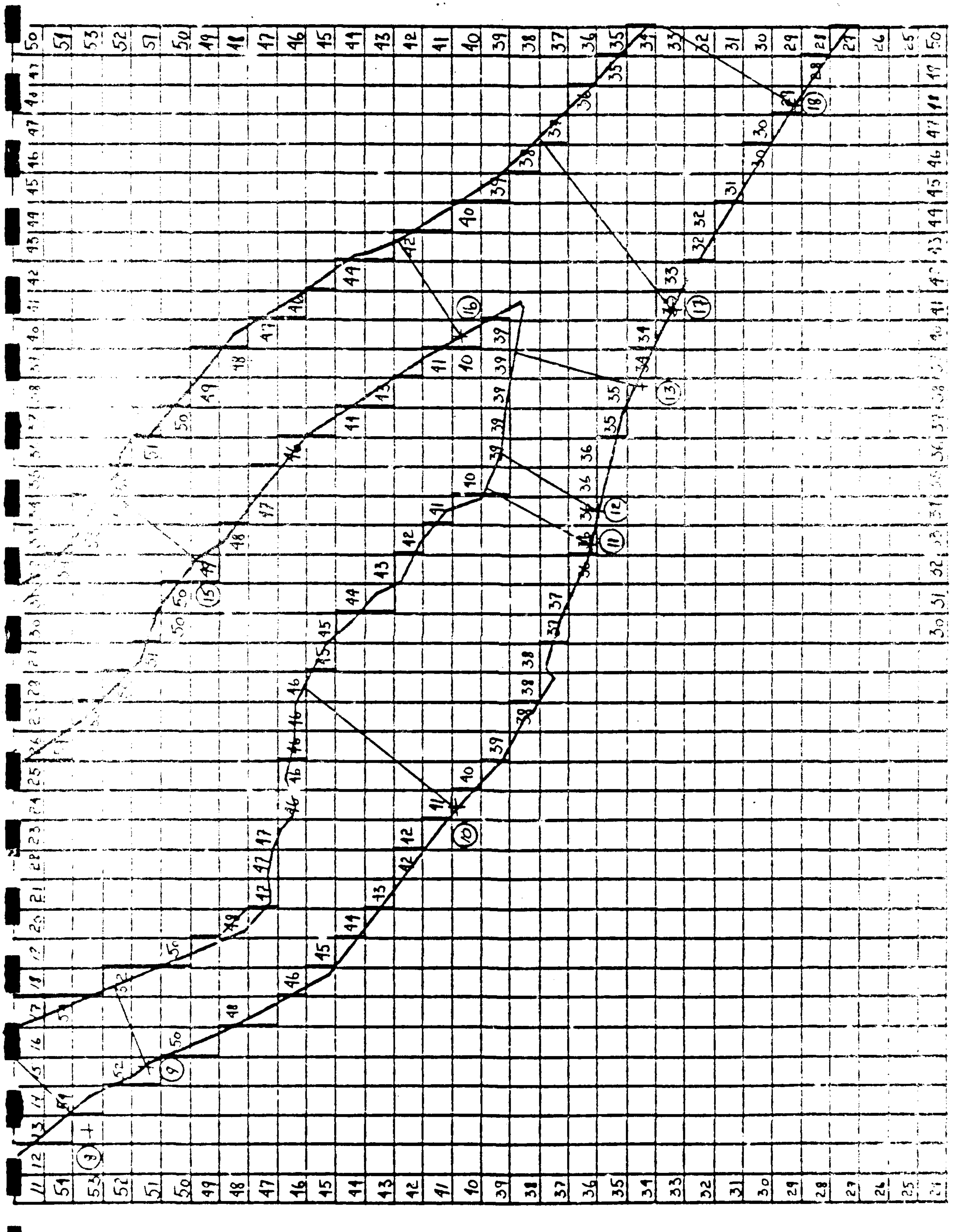


Chart DT2

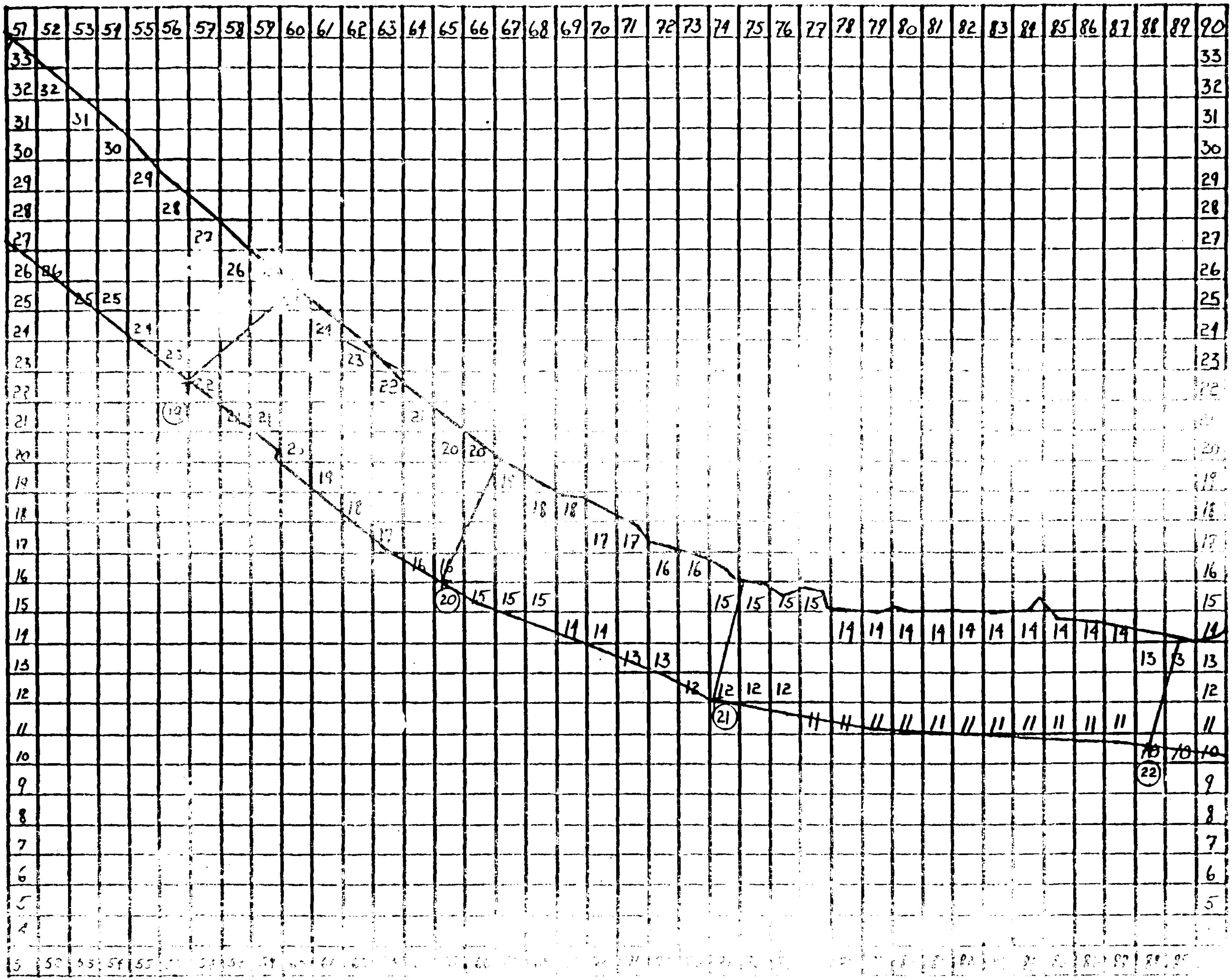


Chart 3

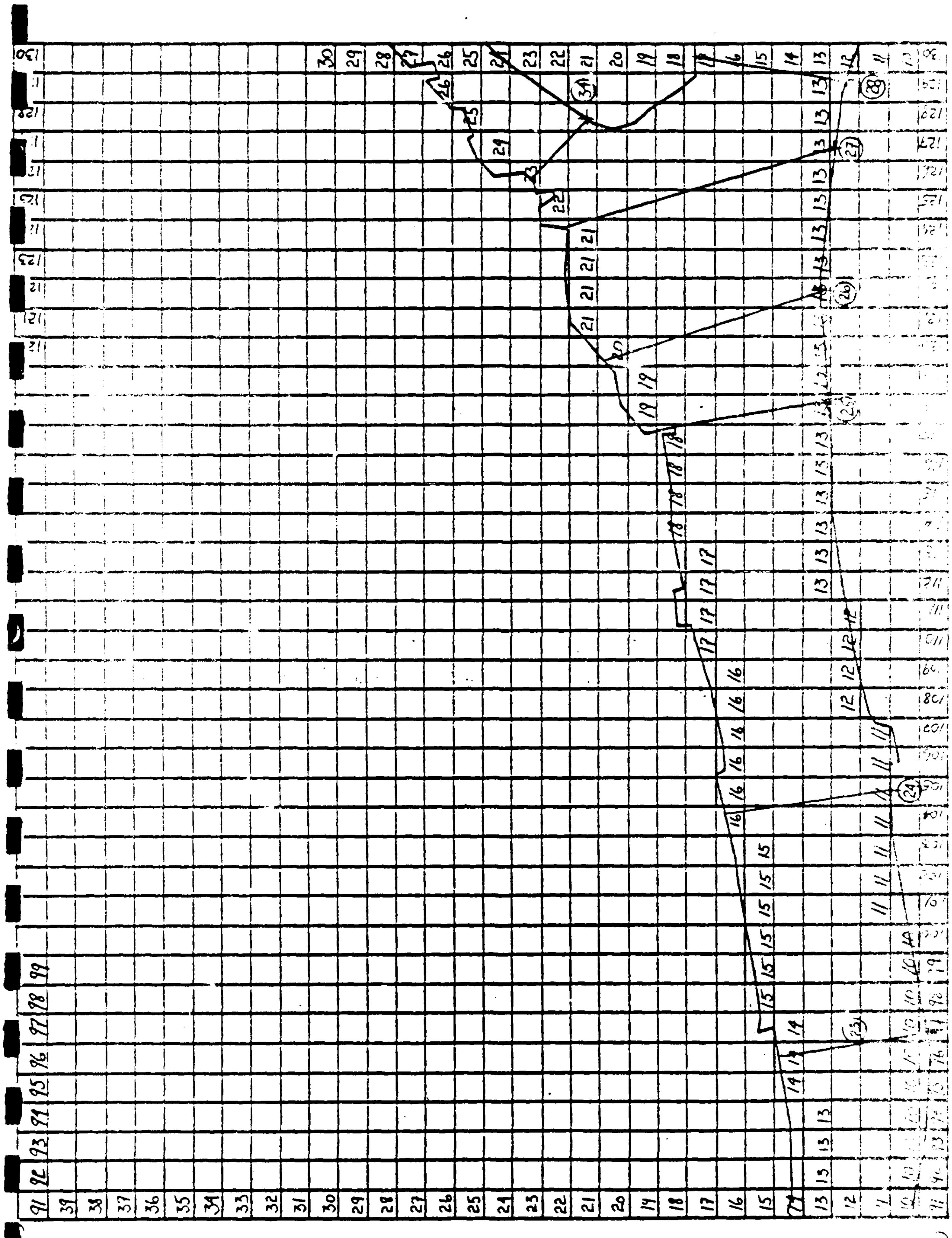


Chart DT4

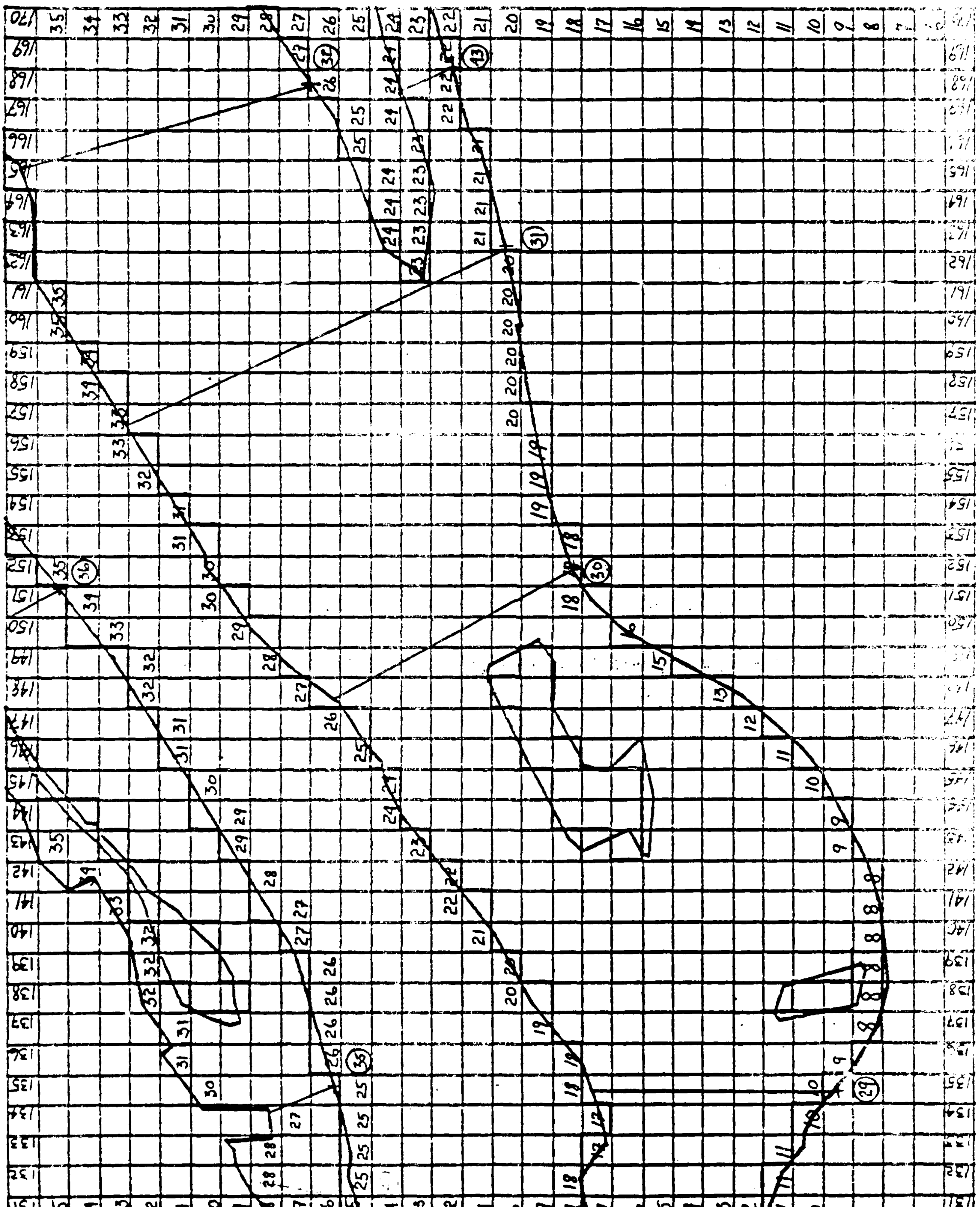


Chart DT5

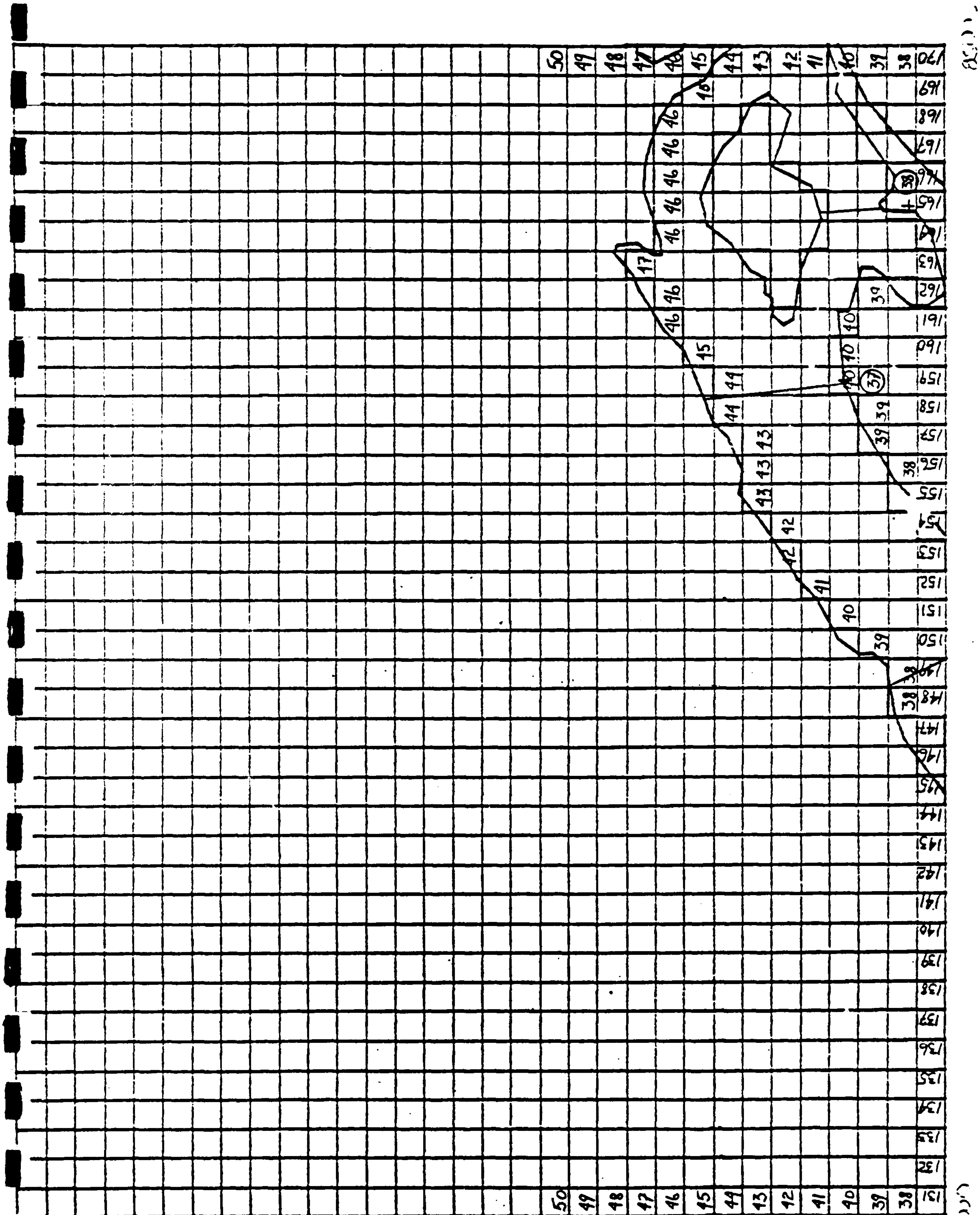
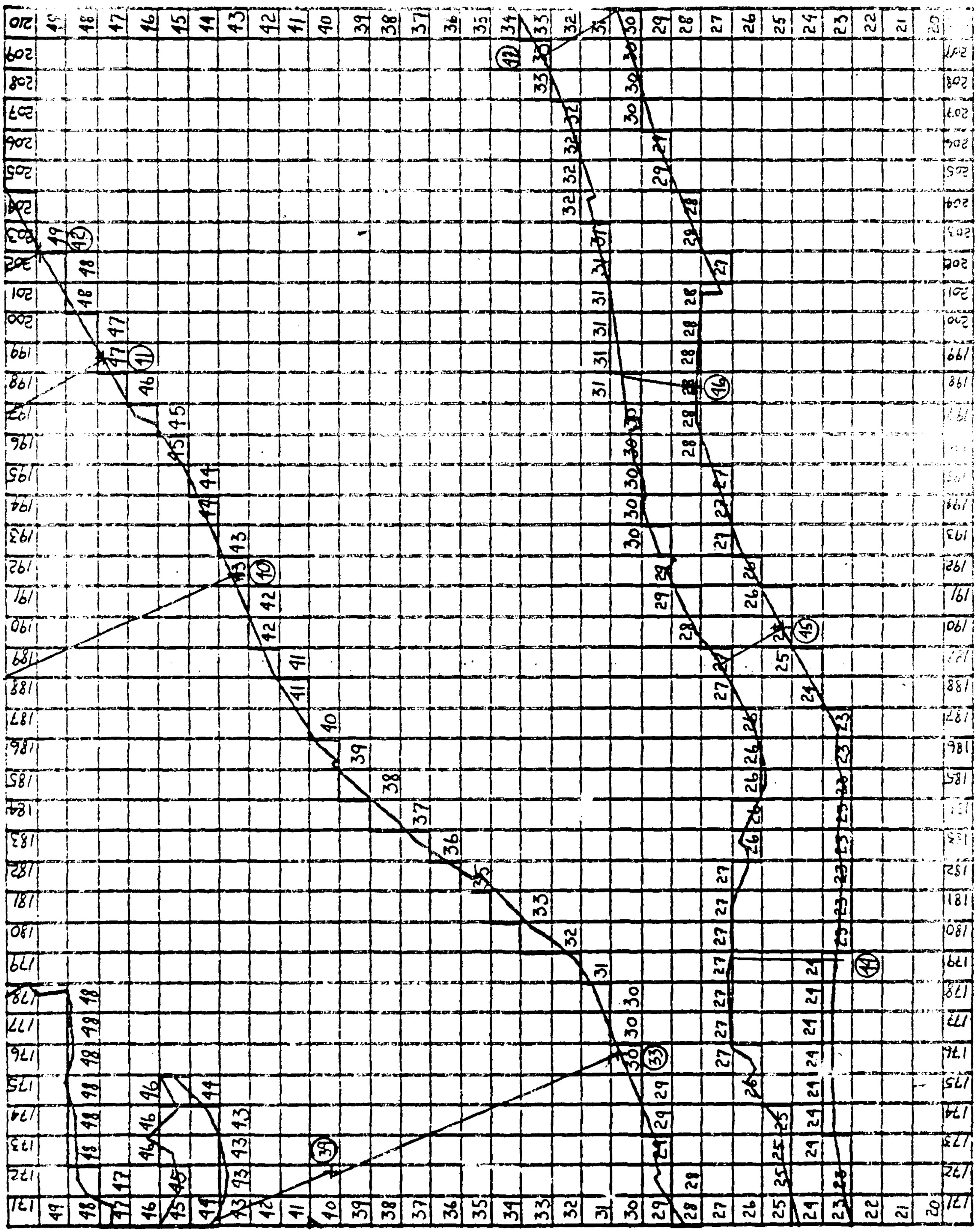


Chart DT6



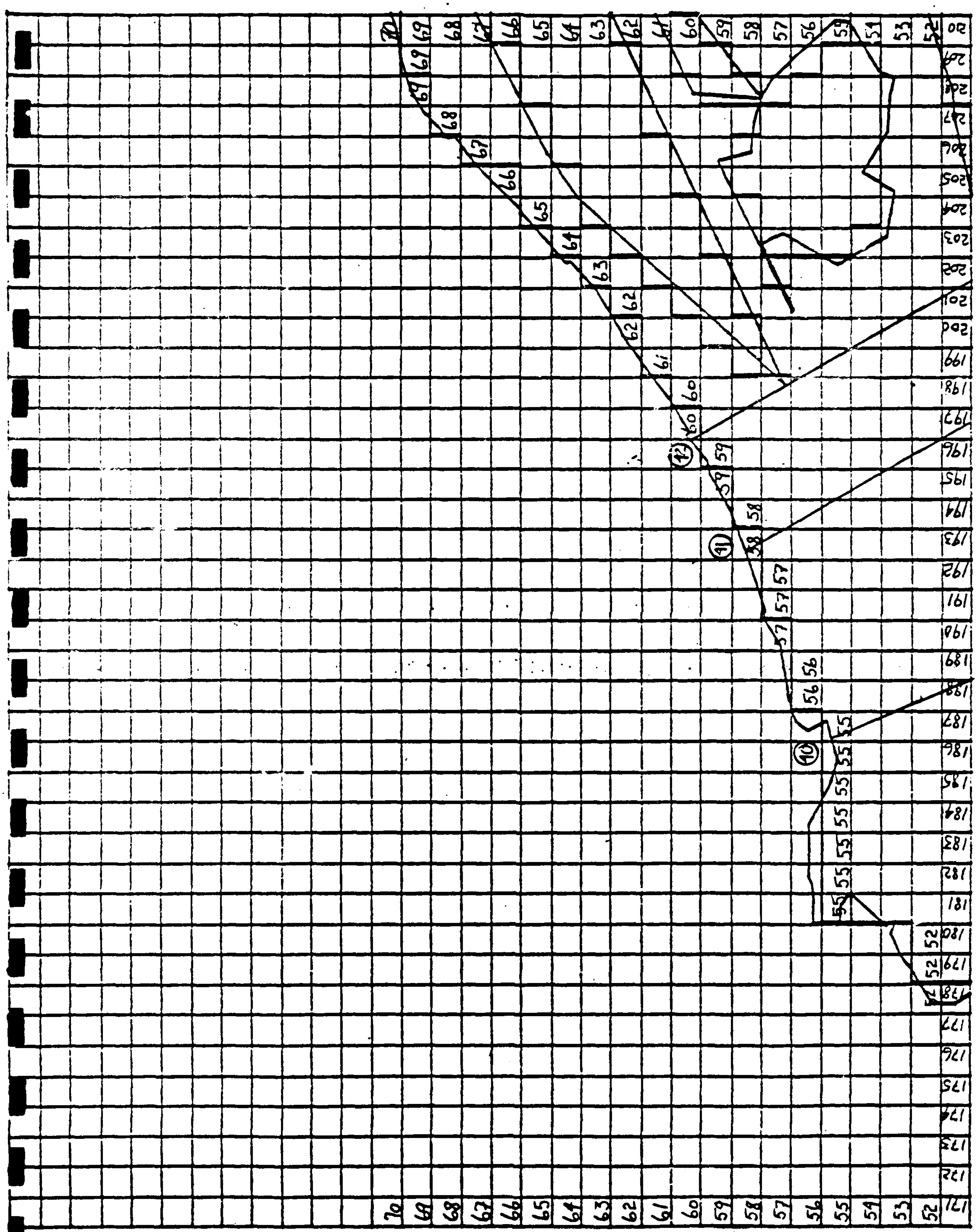
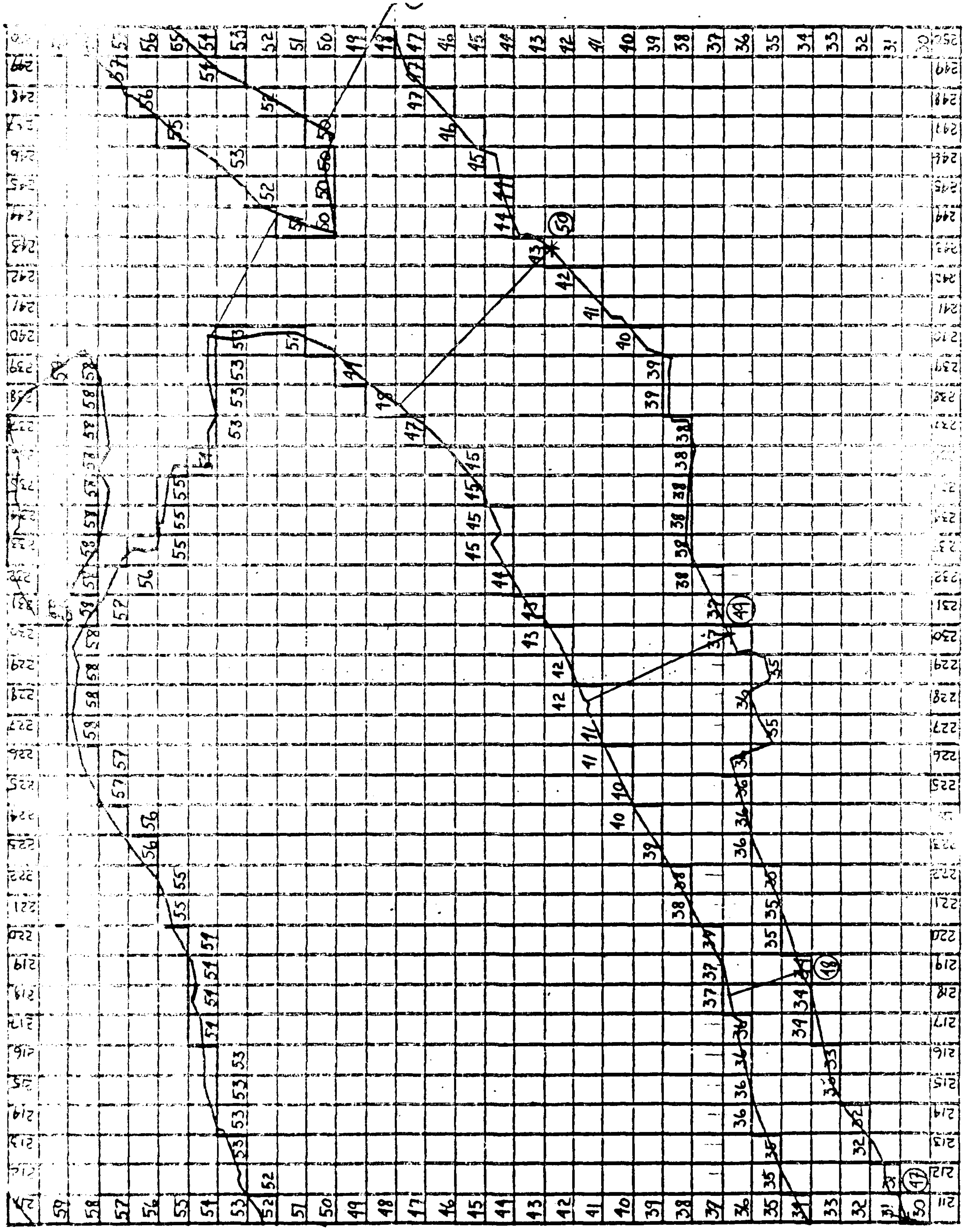


Chart DT8



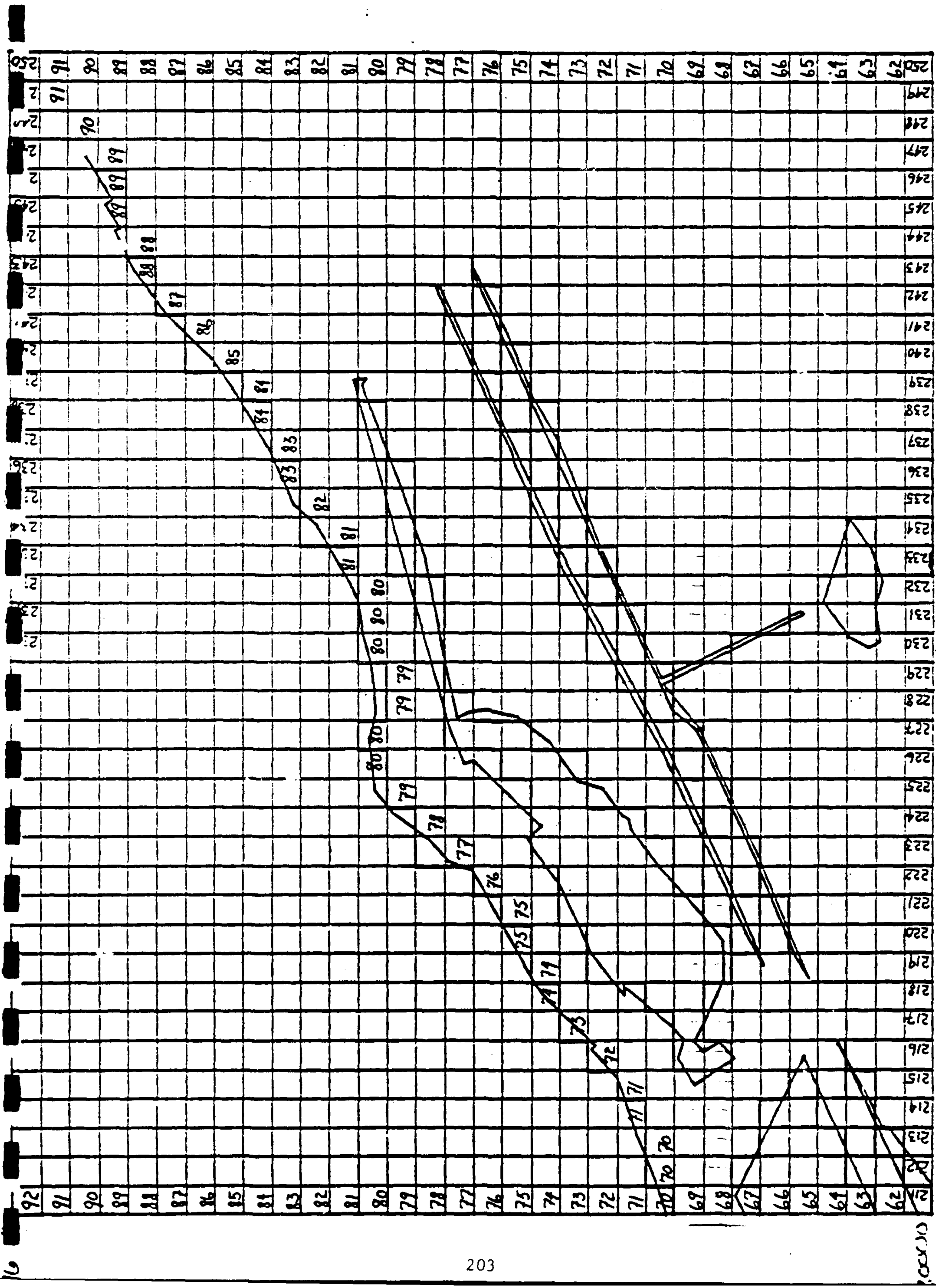
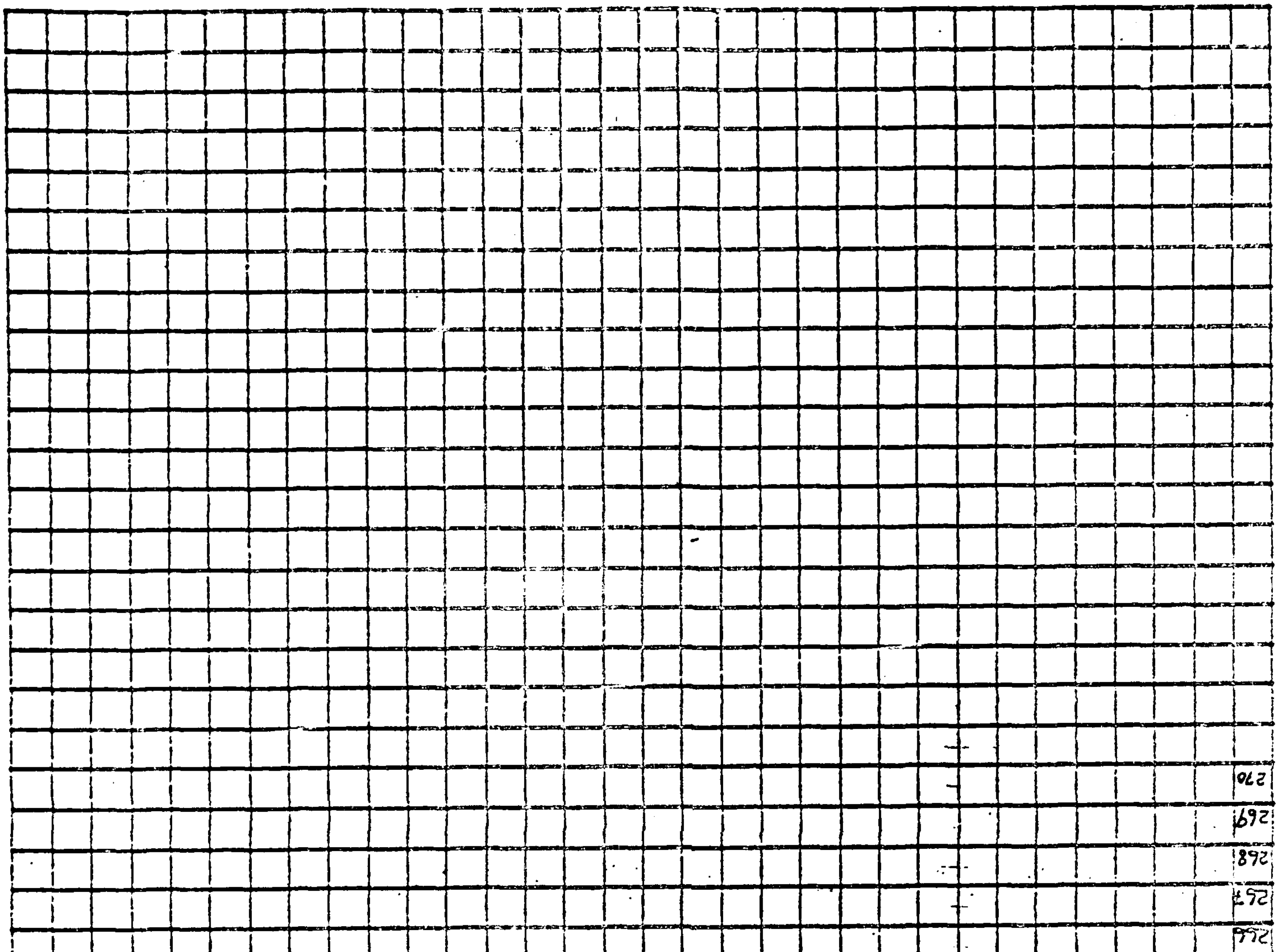
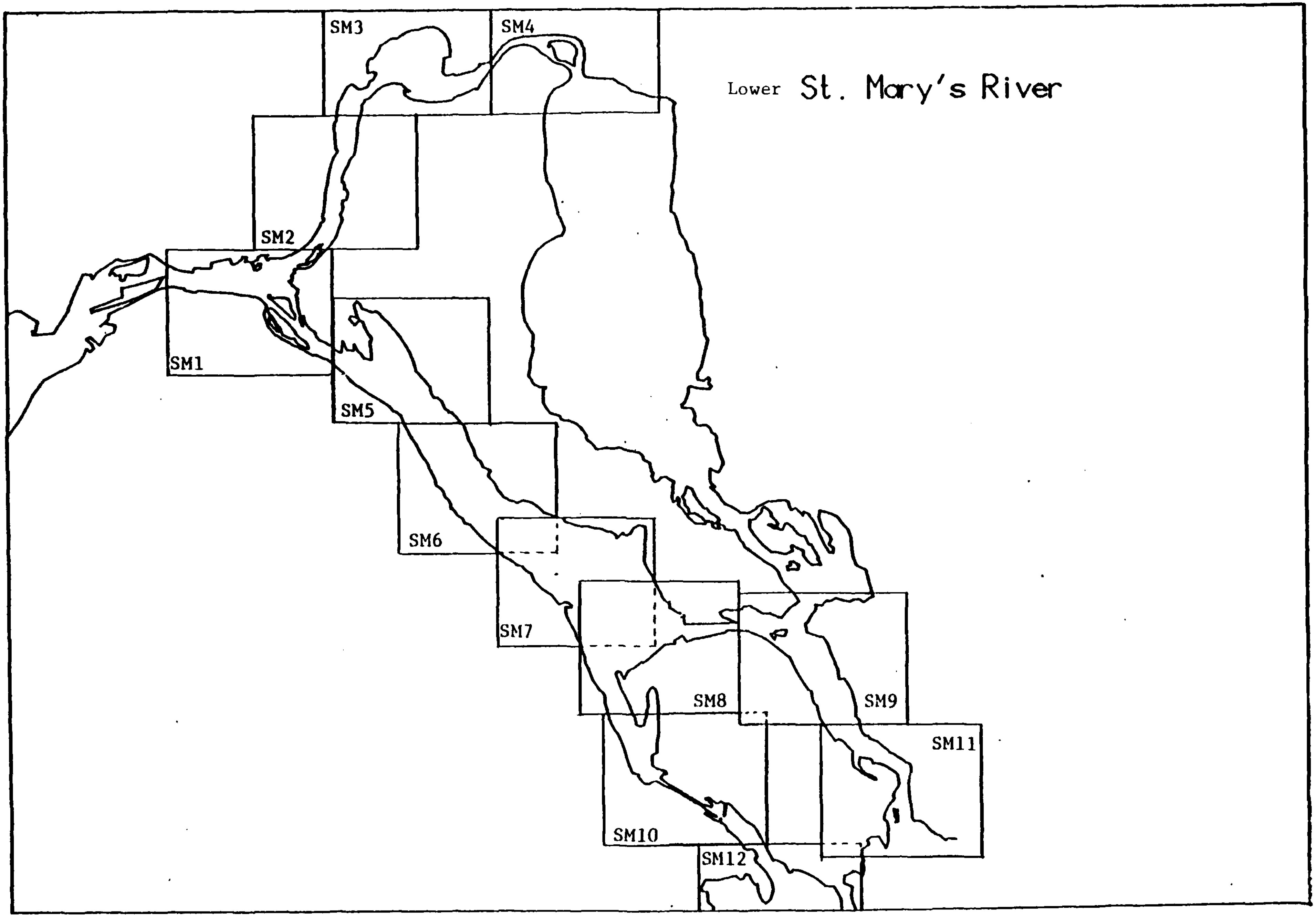


Chart DT10





Index Map for Grid System in Lower St. Mary's River

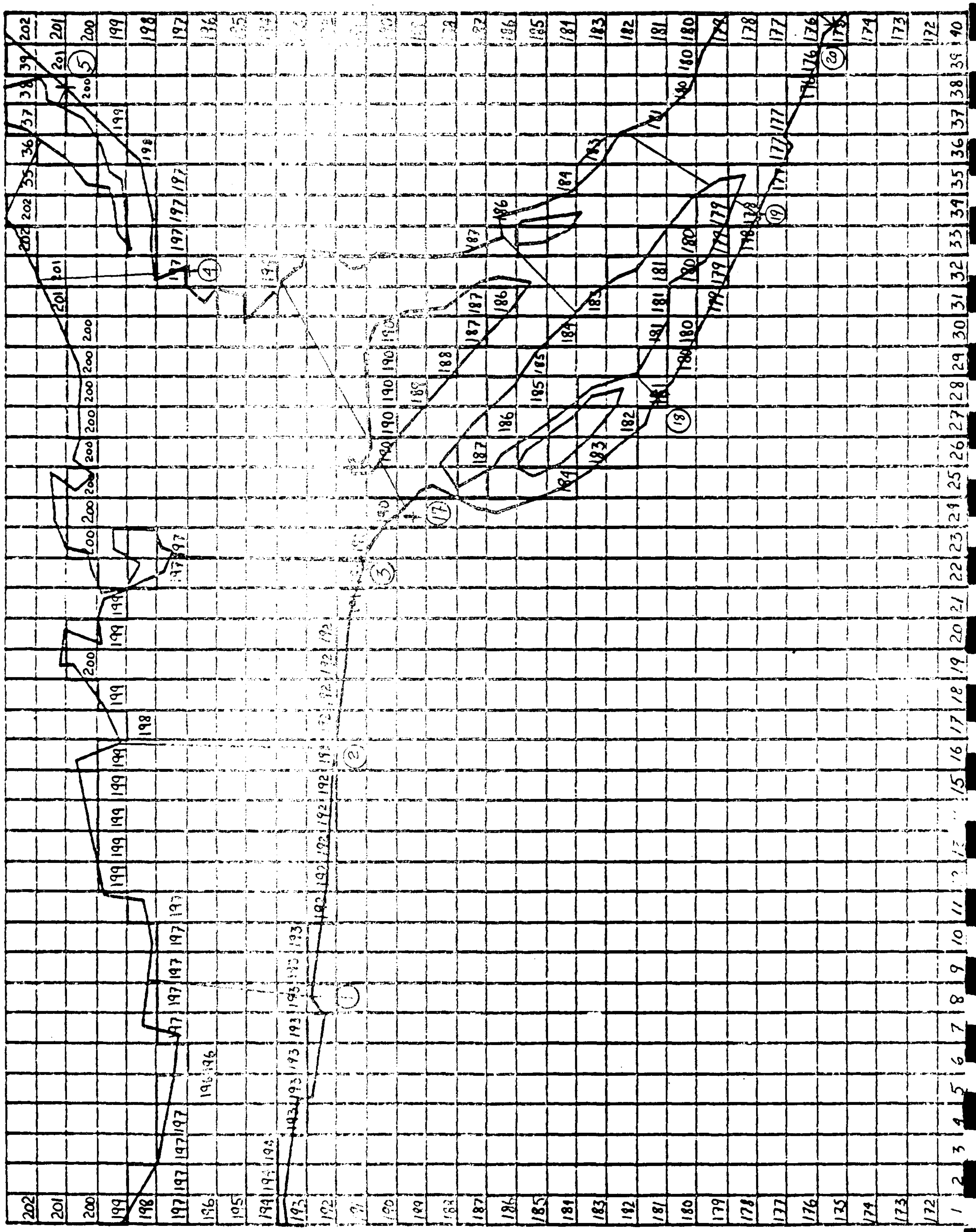


Chart SM1

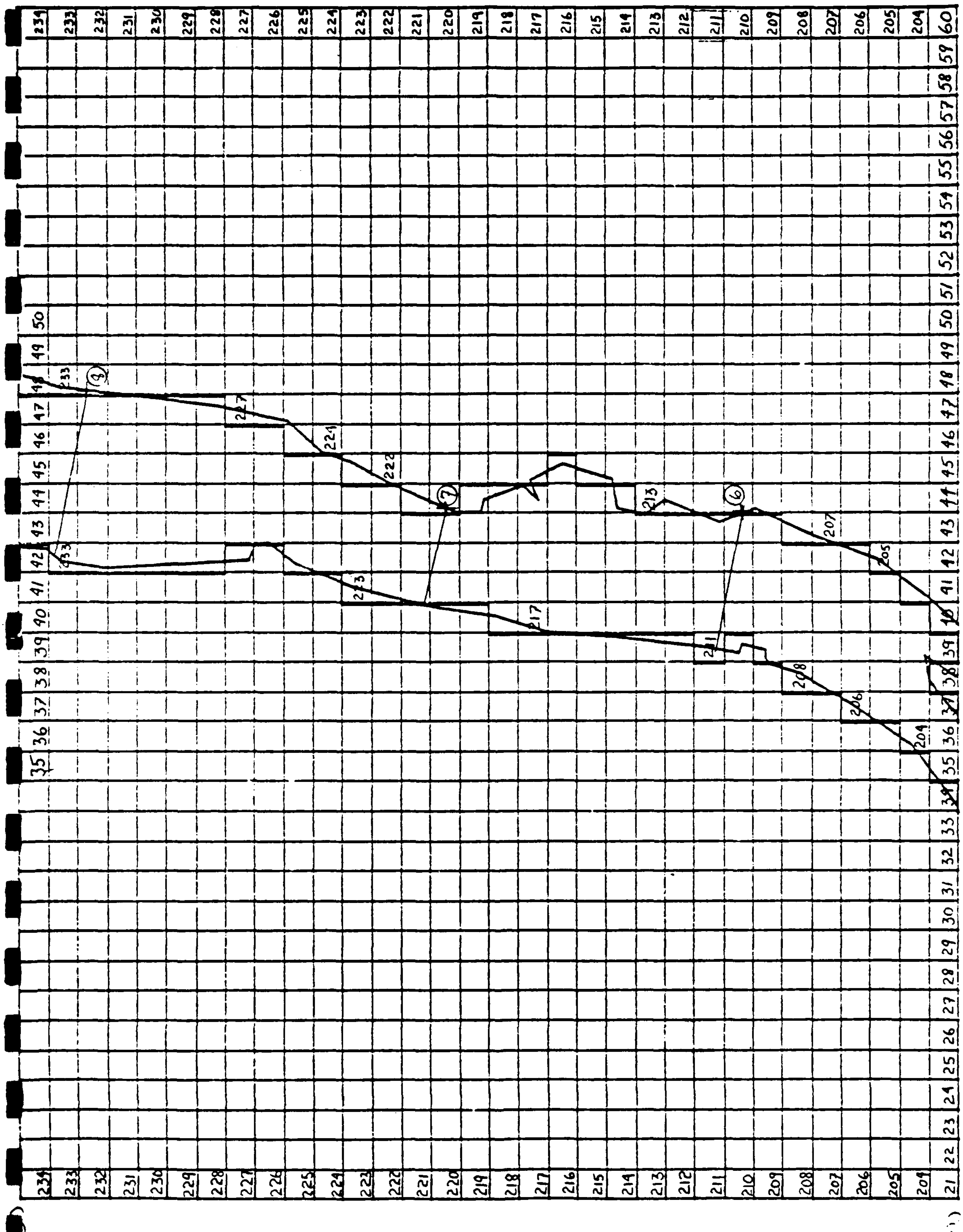
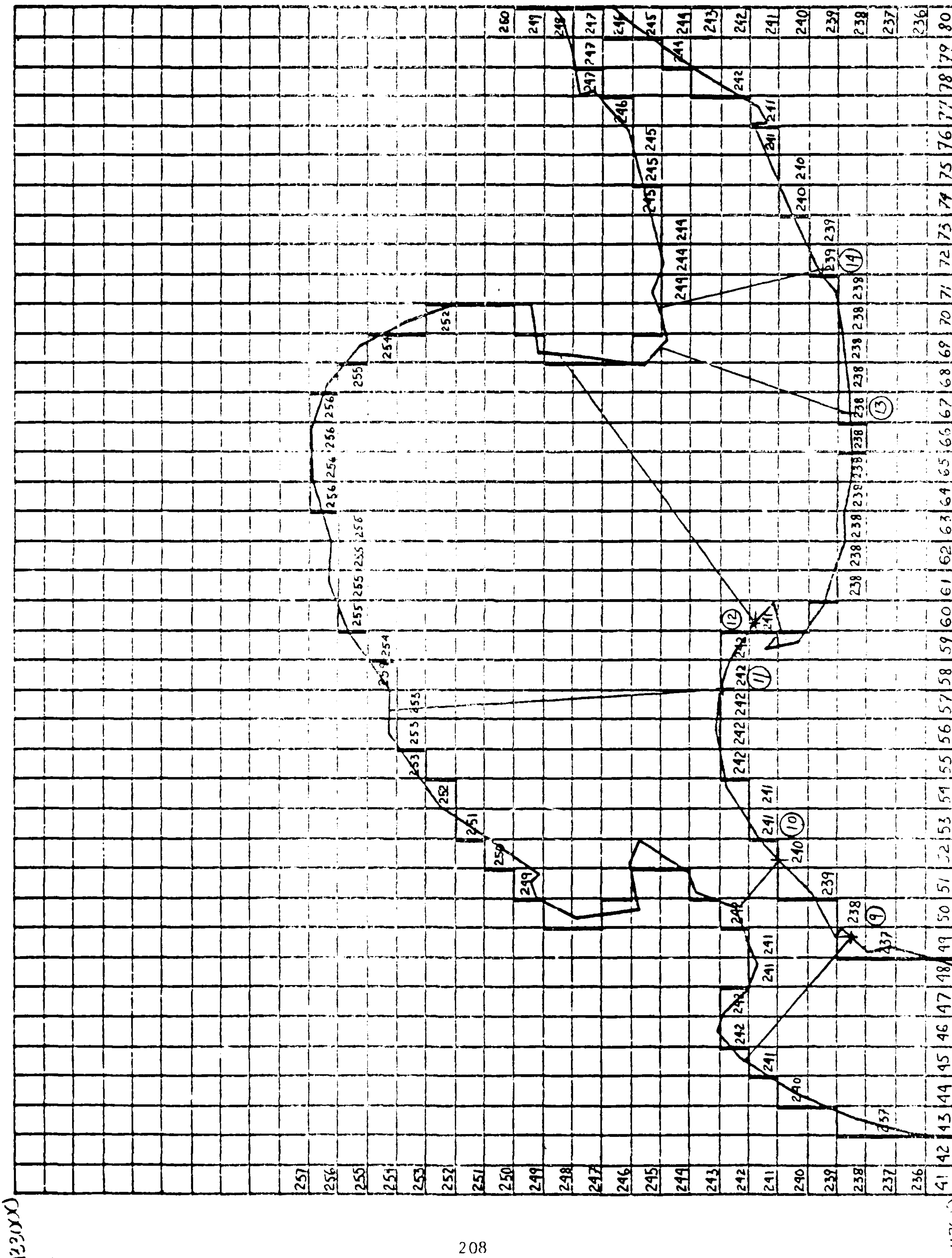


Chart SM2



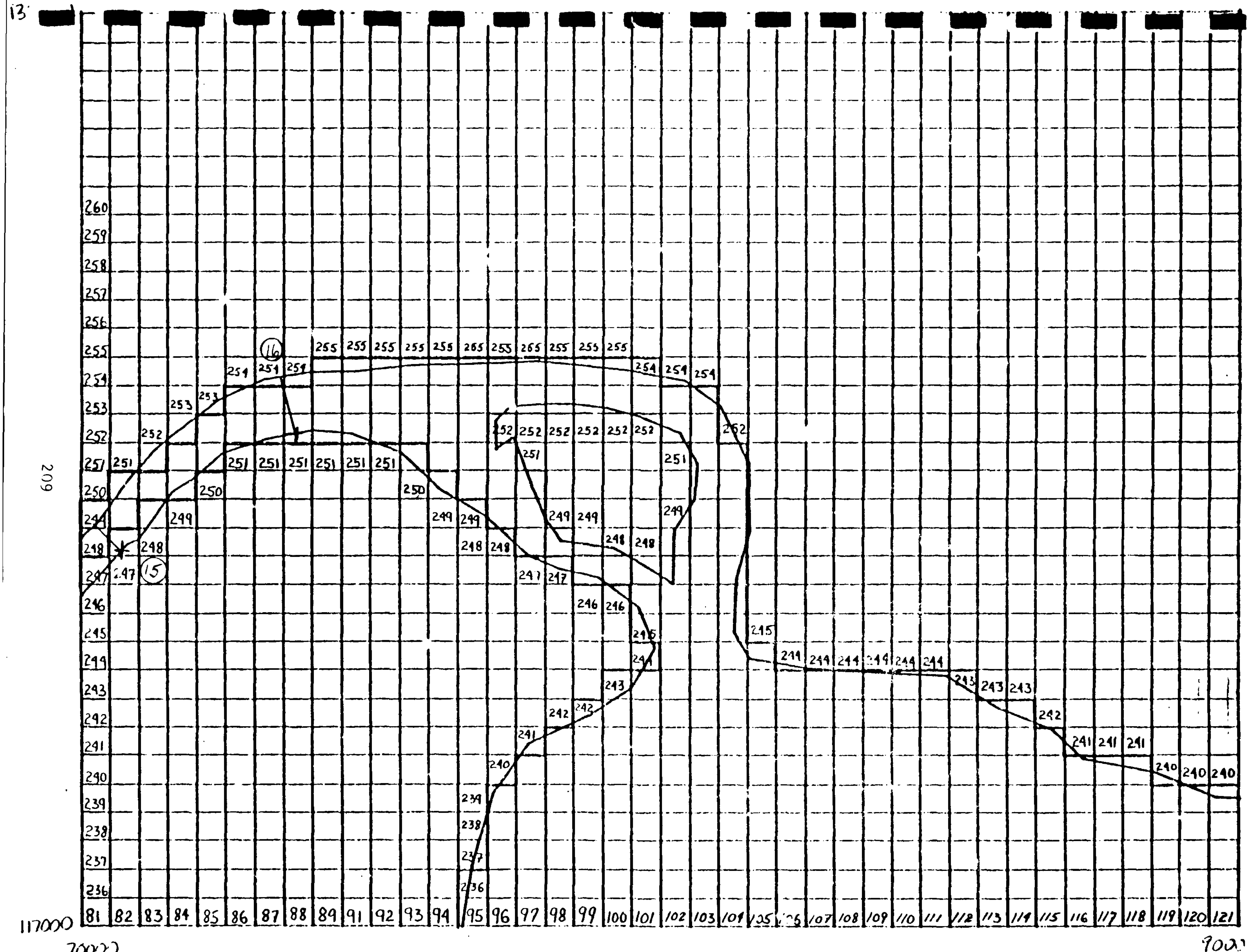
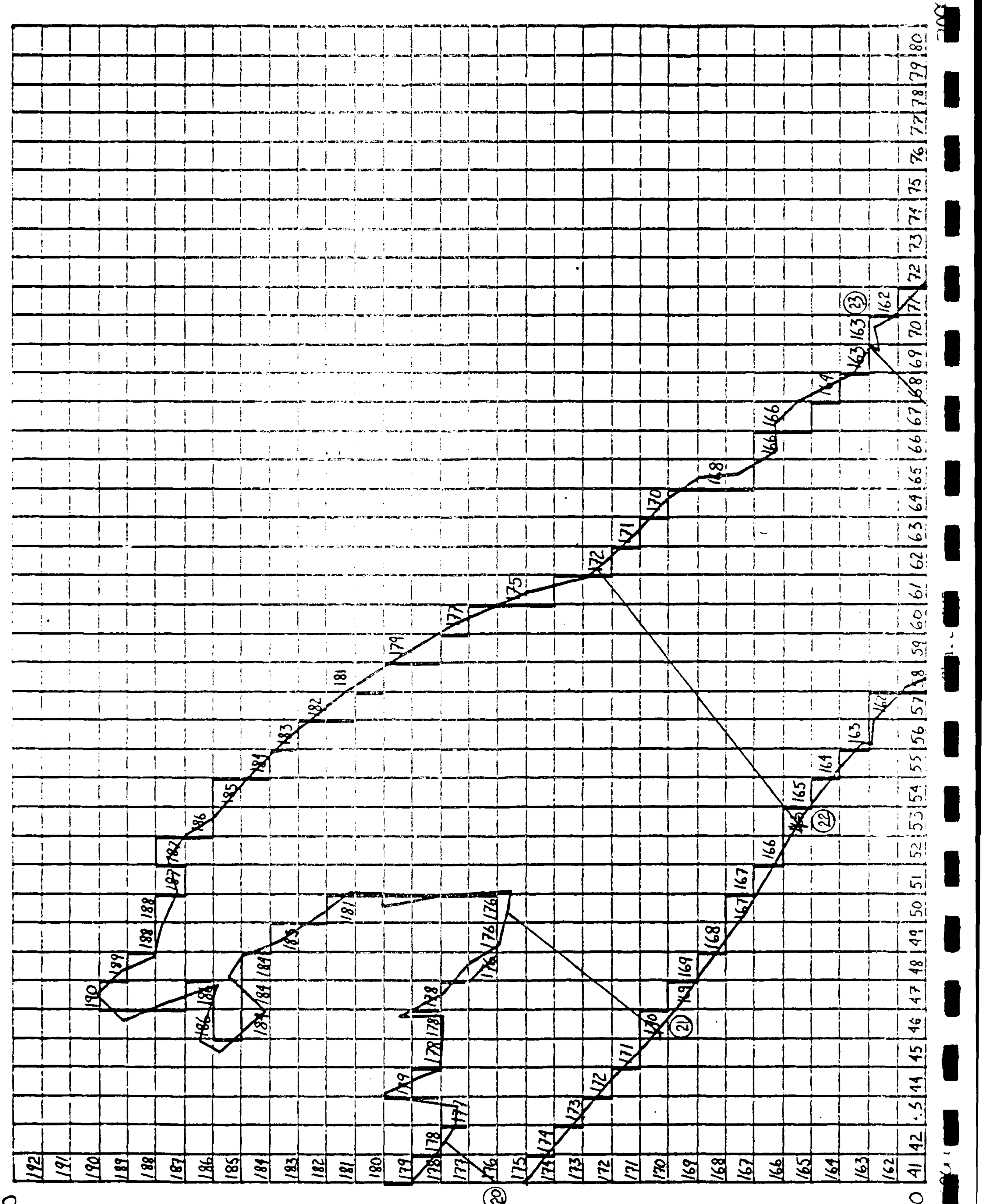


Chart SM4



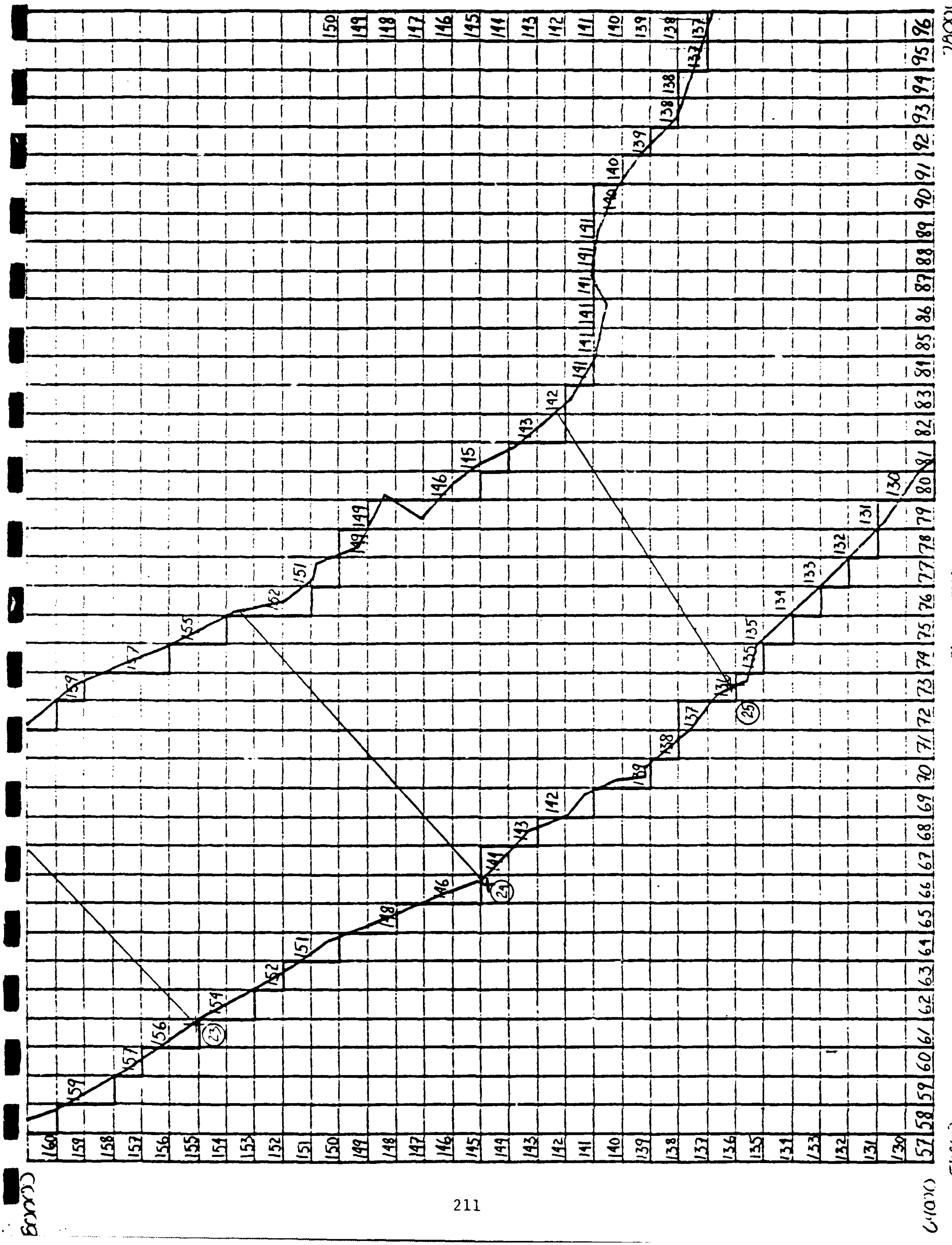
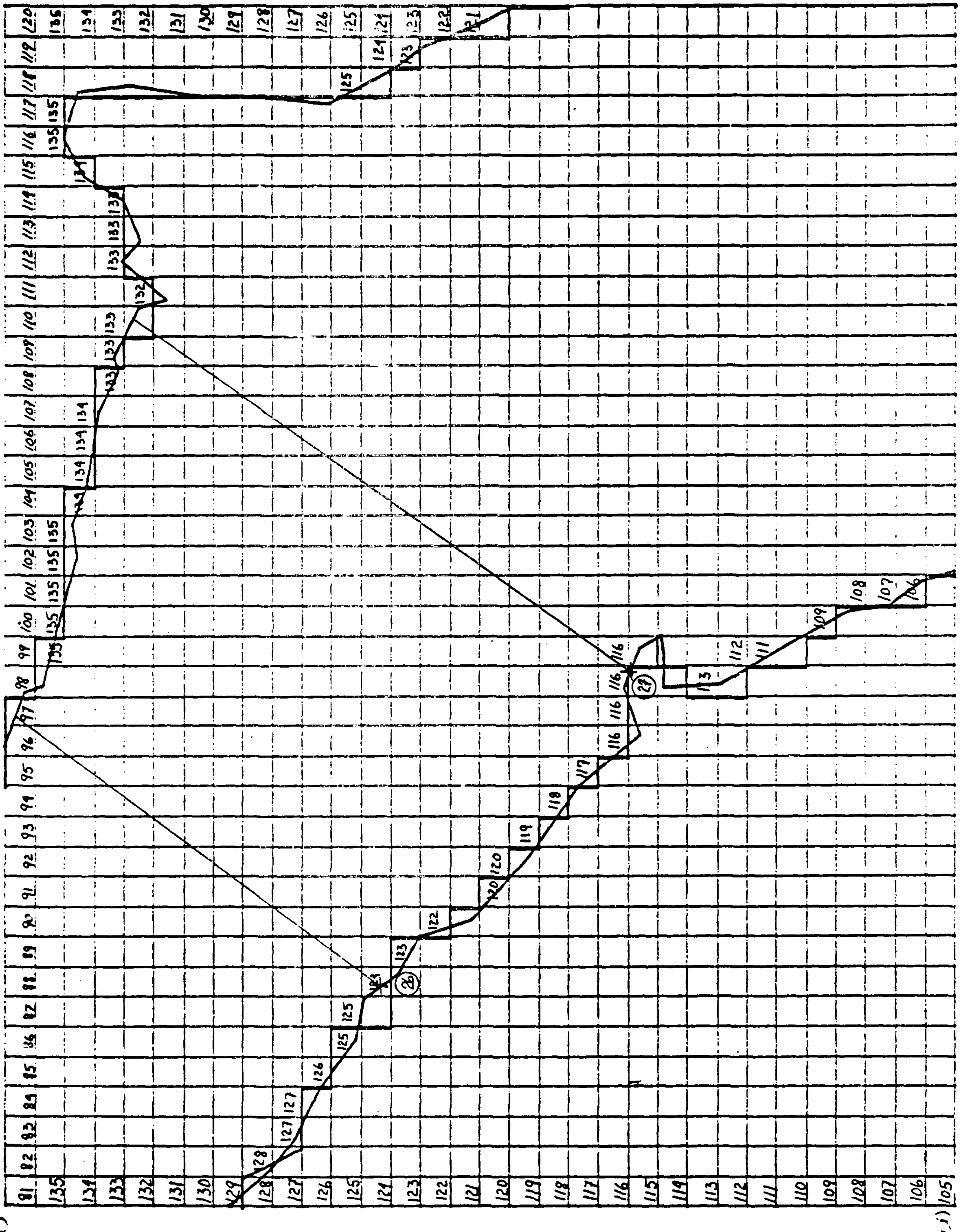


Chart SM6



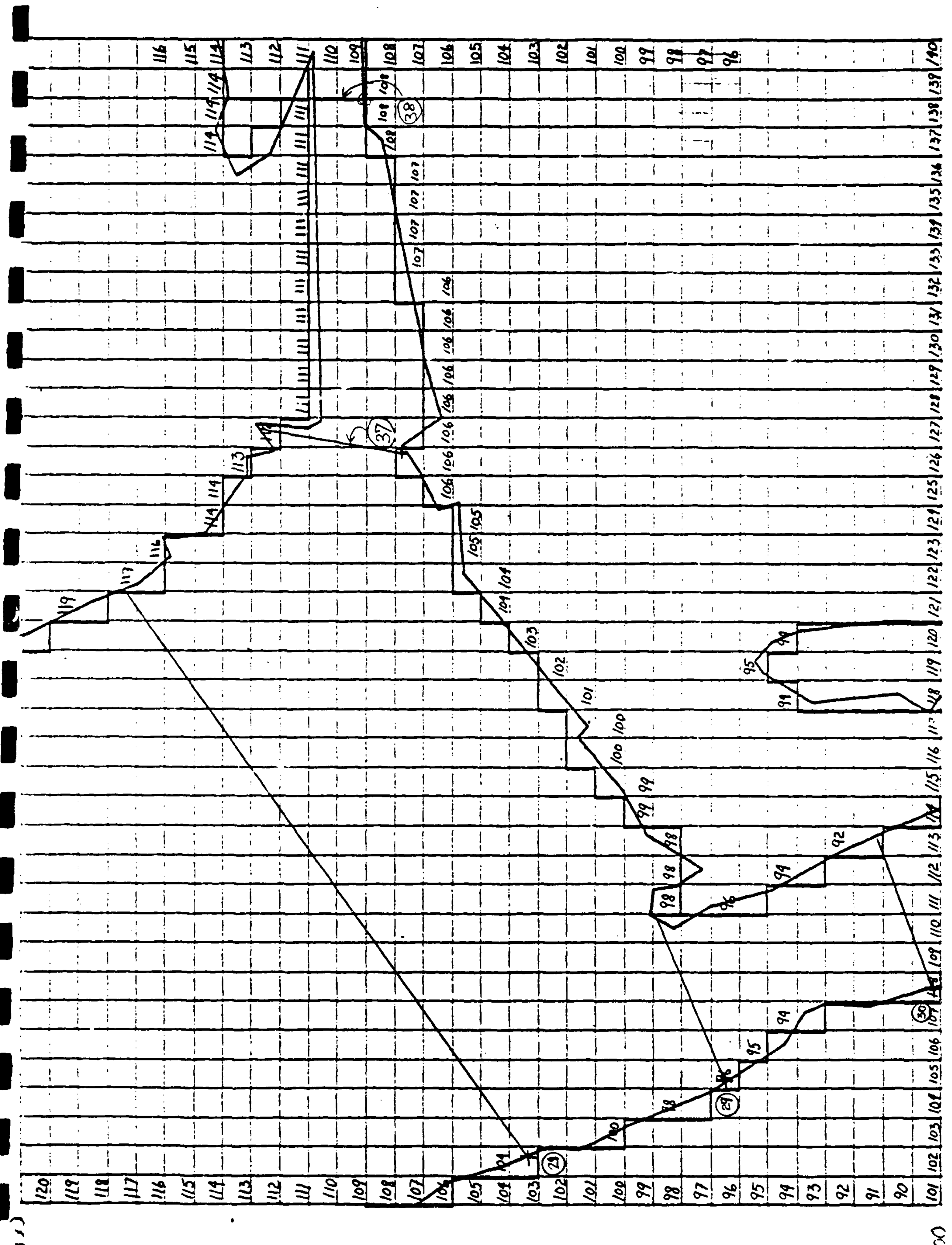
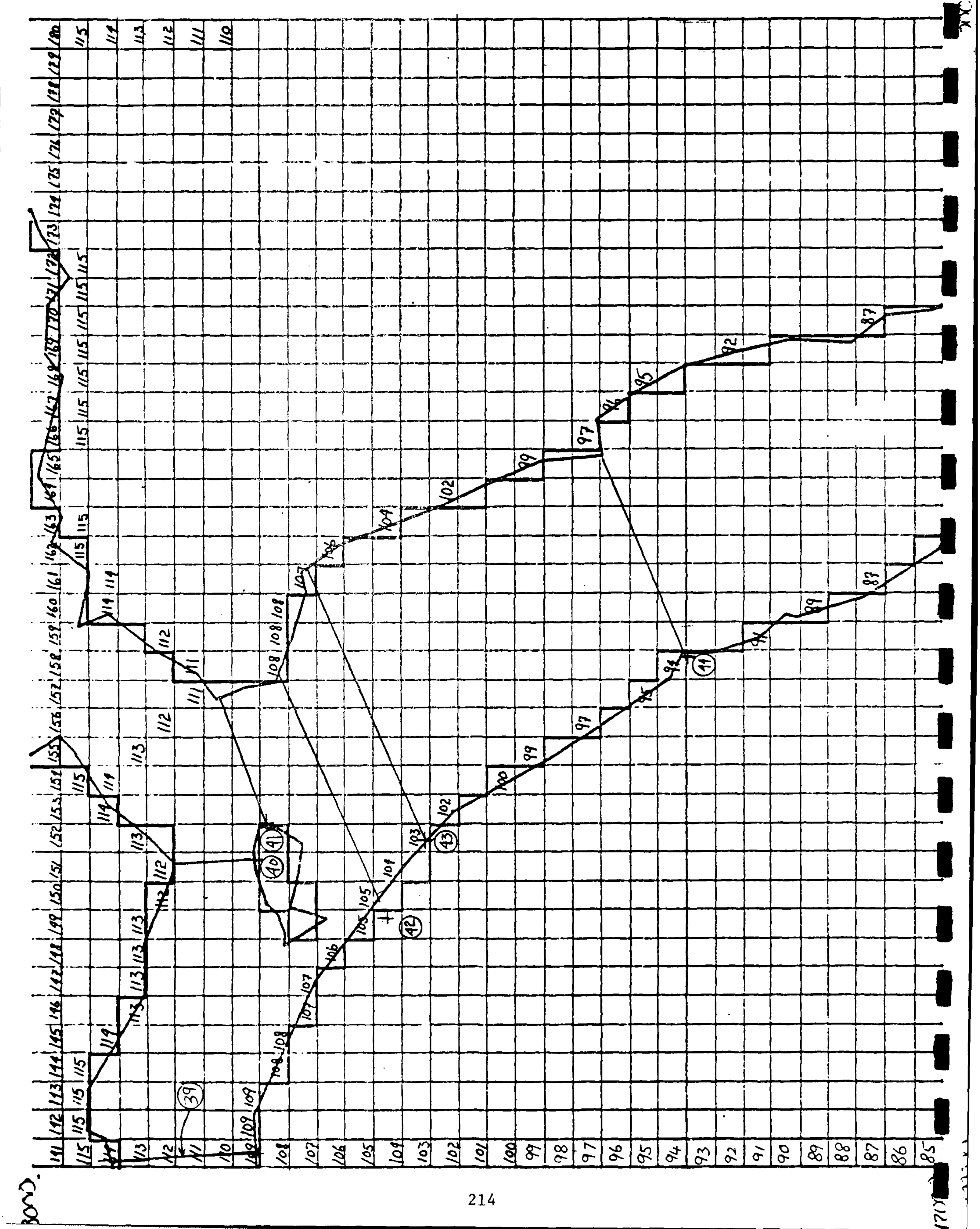


Chart 5M8



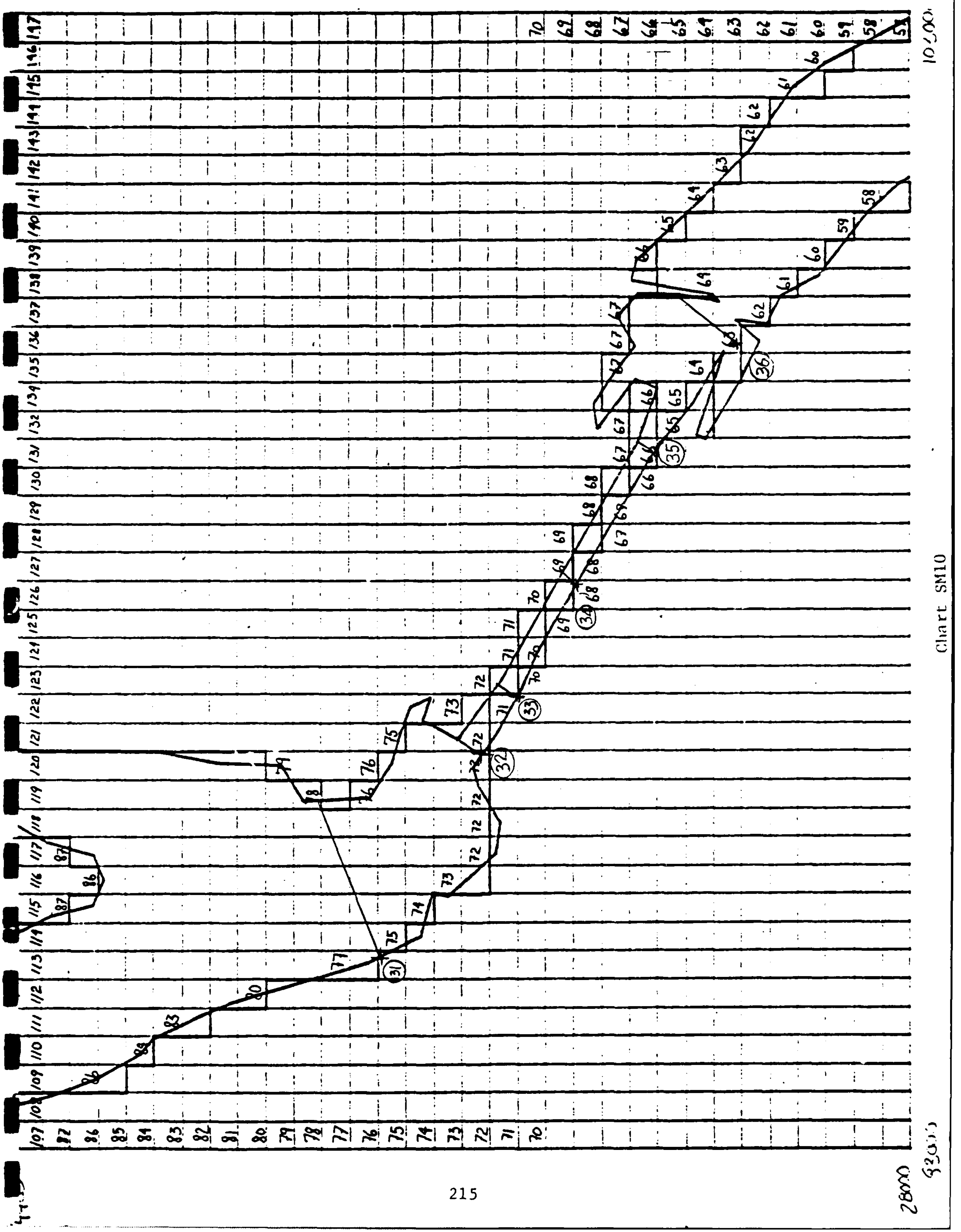
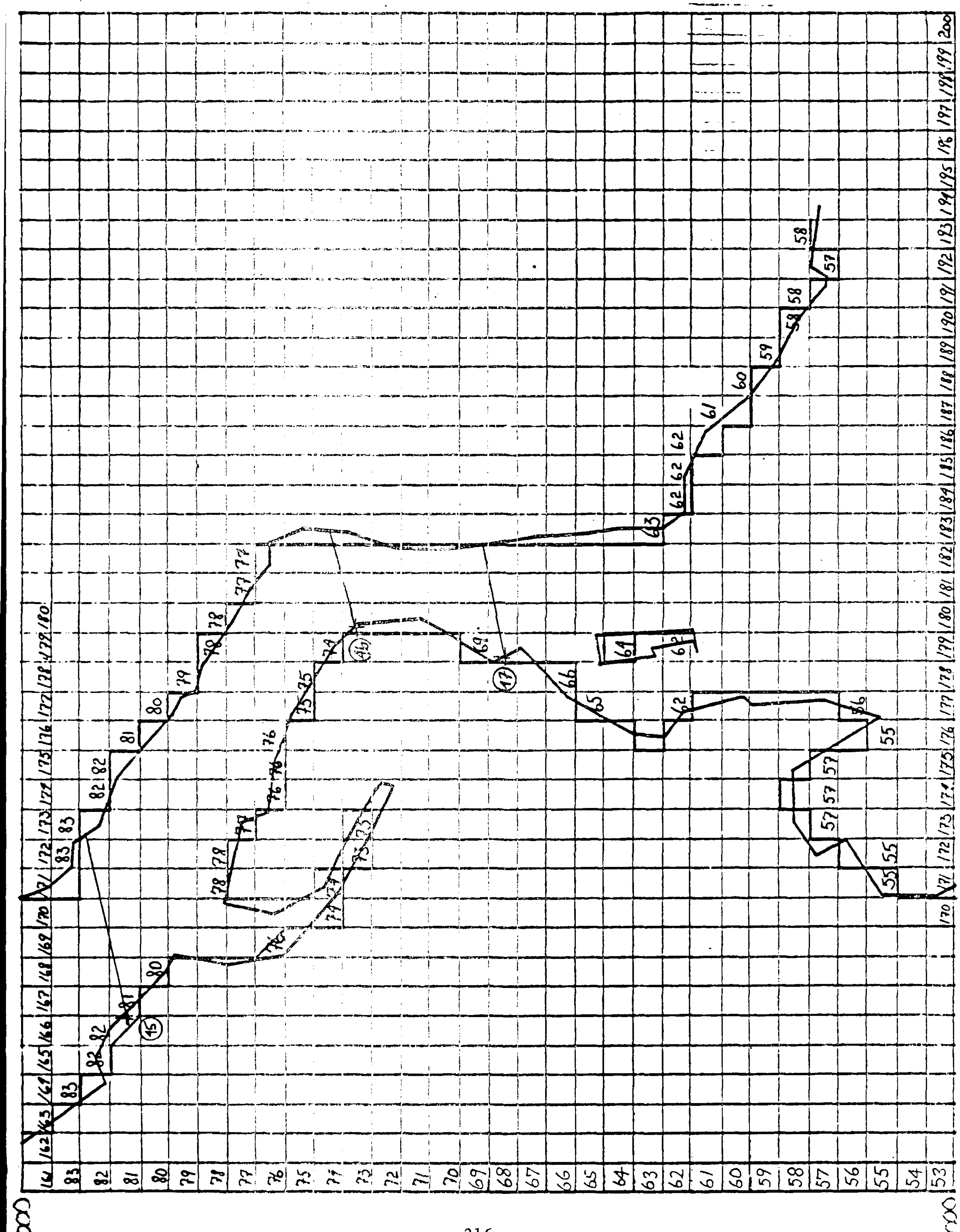


Chart SM10



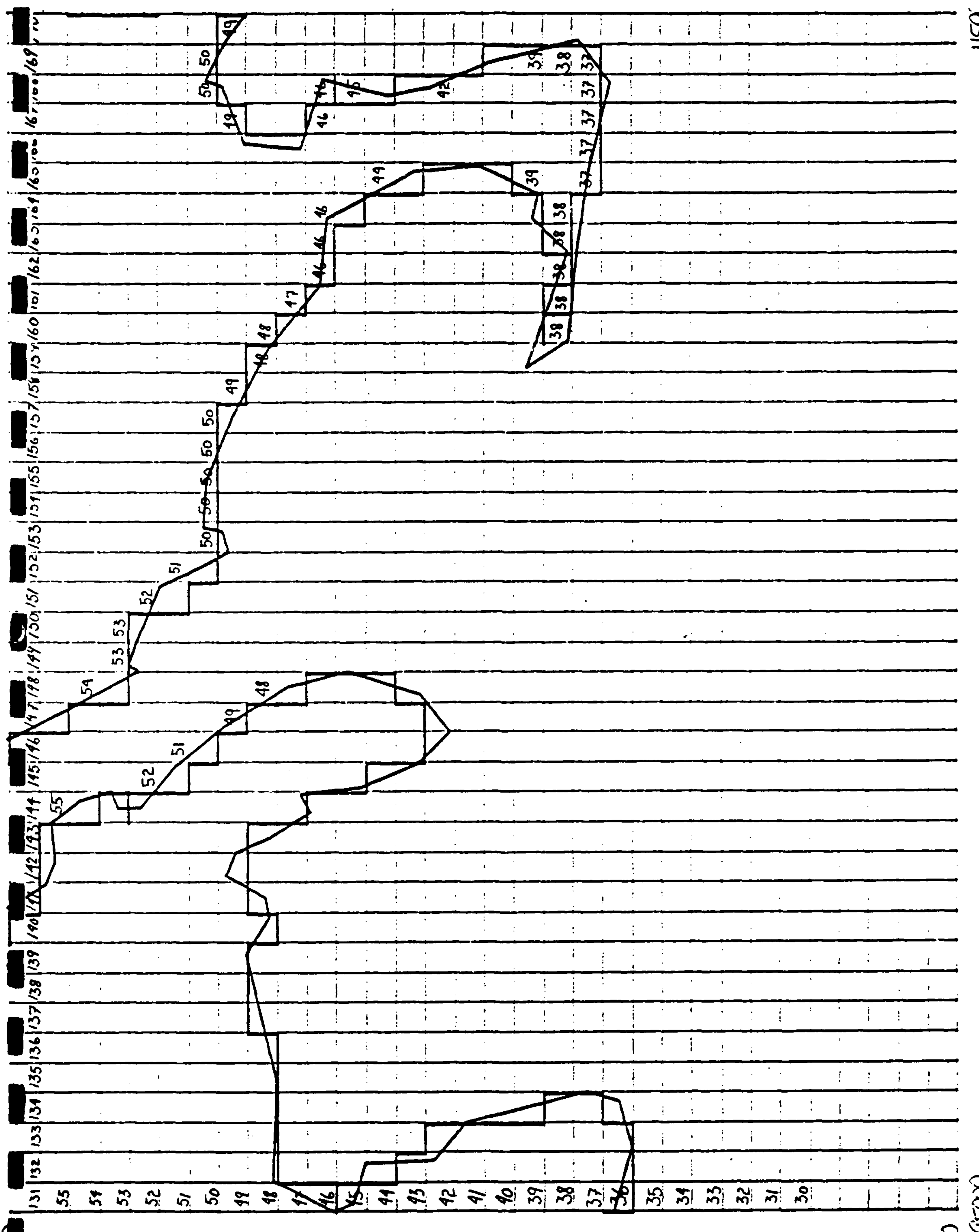
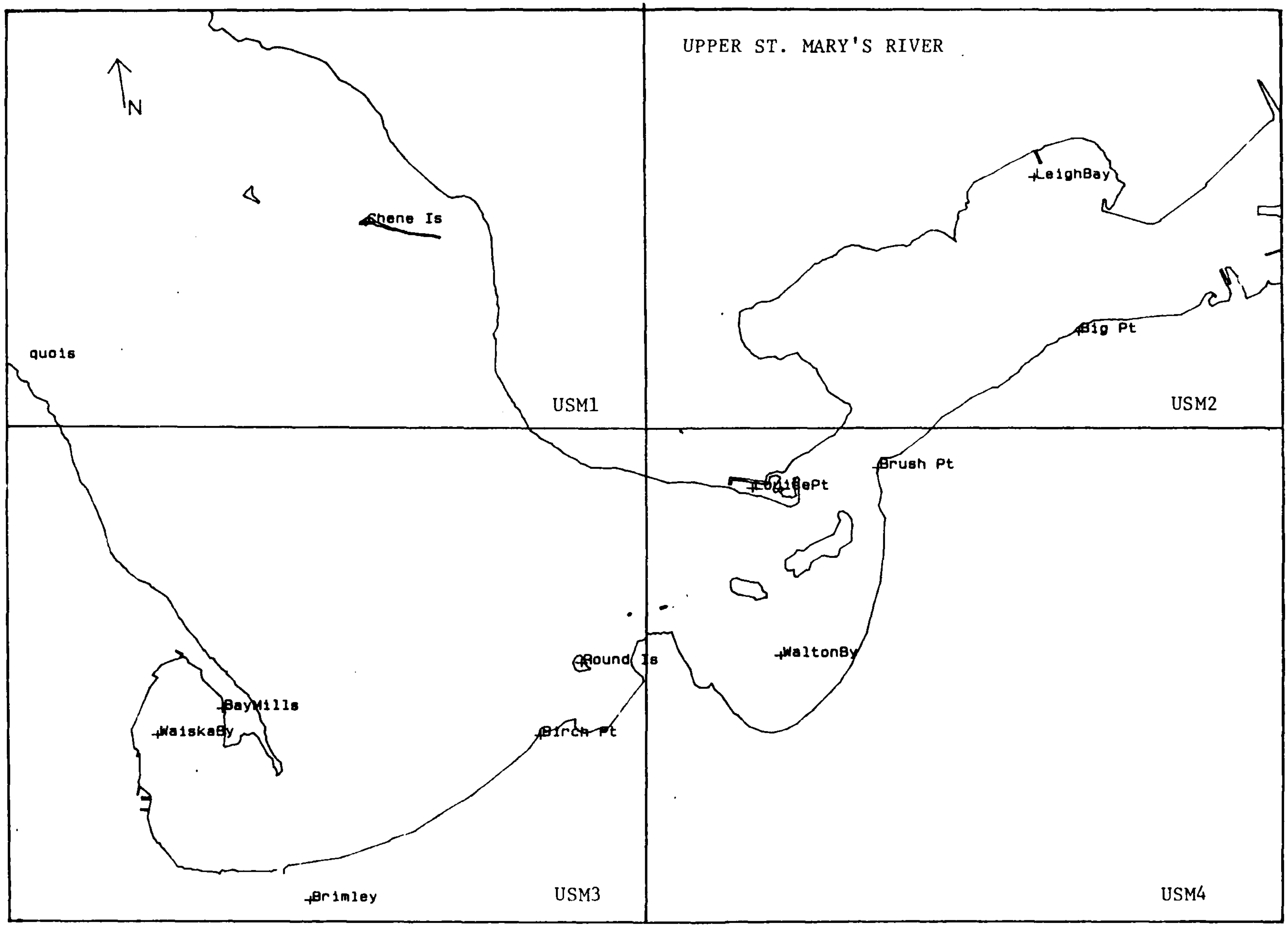

55
54
53
52
51
50
49
48
47
46
45
44
43
42
41
40
39
38
37
36
35
34
33
32
31
30
1200
1150

Chart SM12



Index Map for Grid System in Upper St. Mary's River

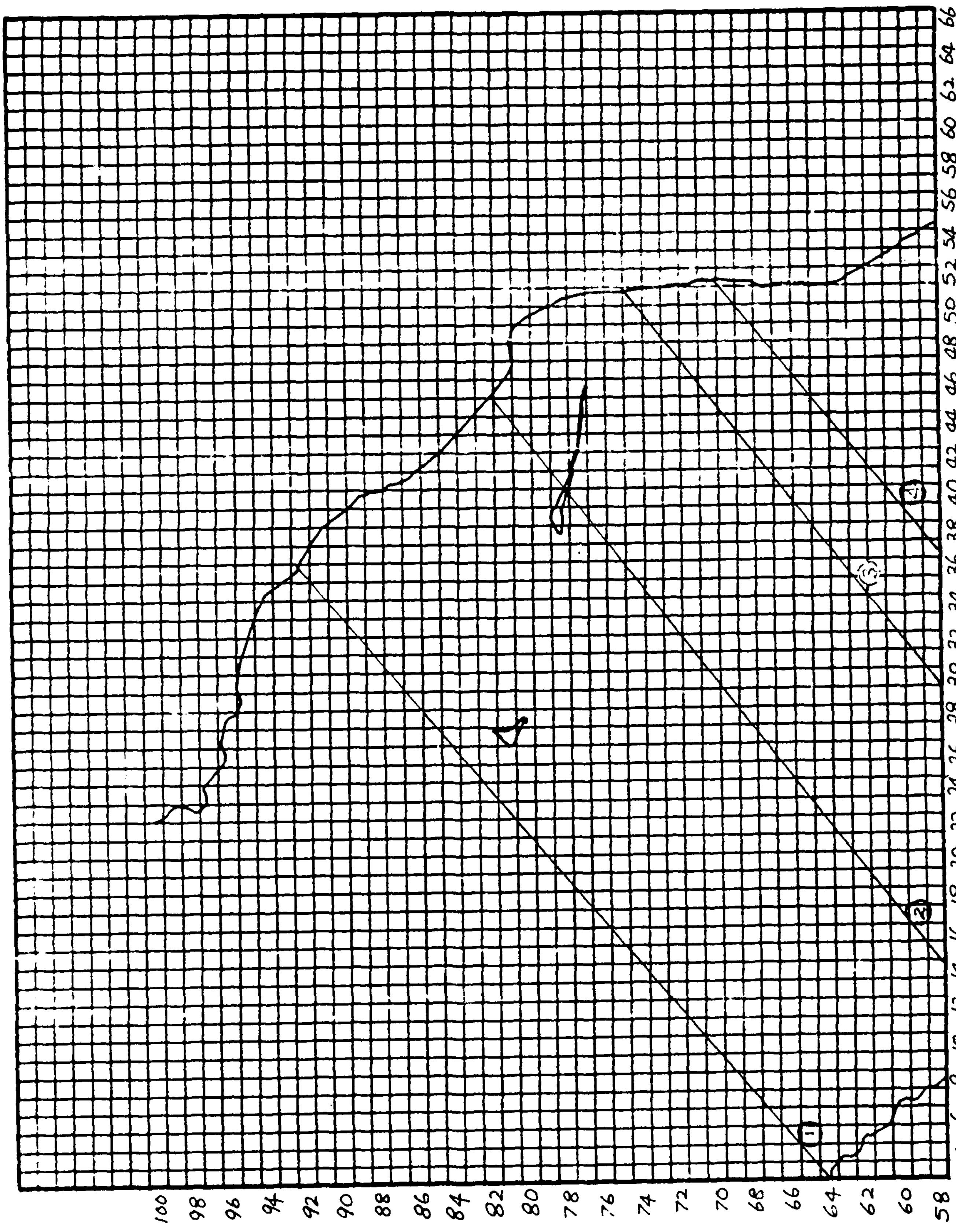

4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66
100 98 96 94 92 90 88 86 84 82 80 78 76 74 72 70 68 66 64 62 60 58
1
2
3
4

Chart USM1

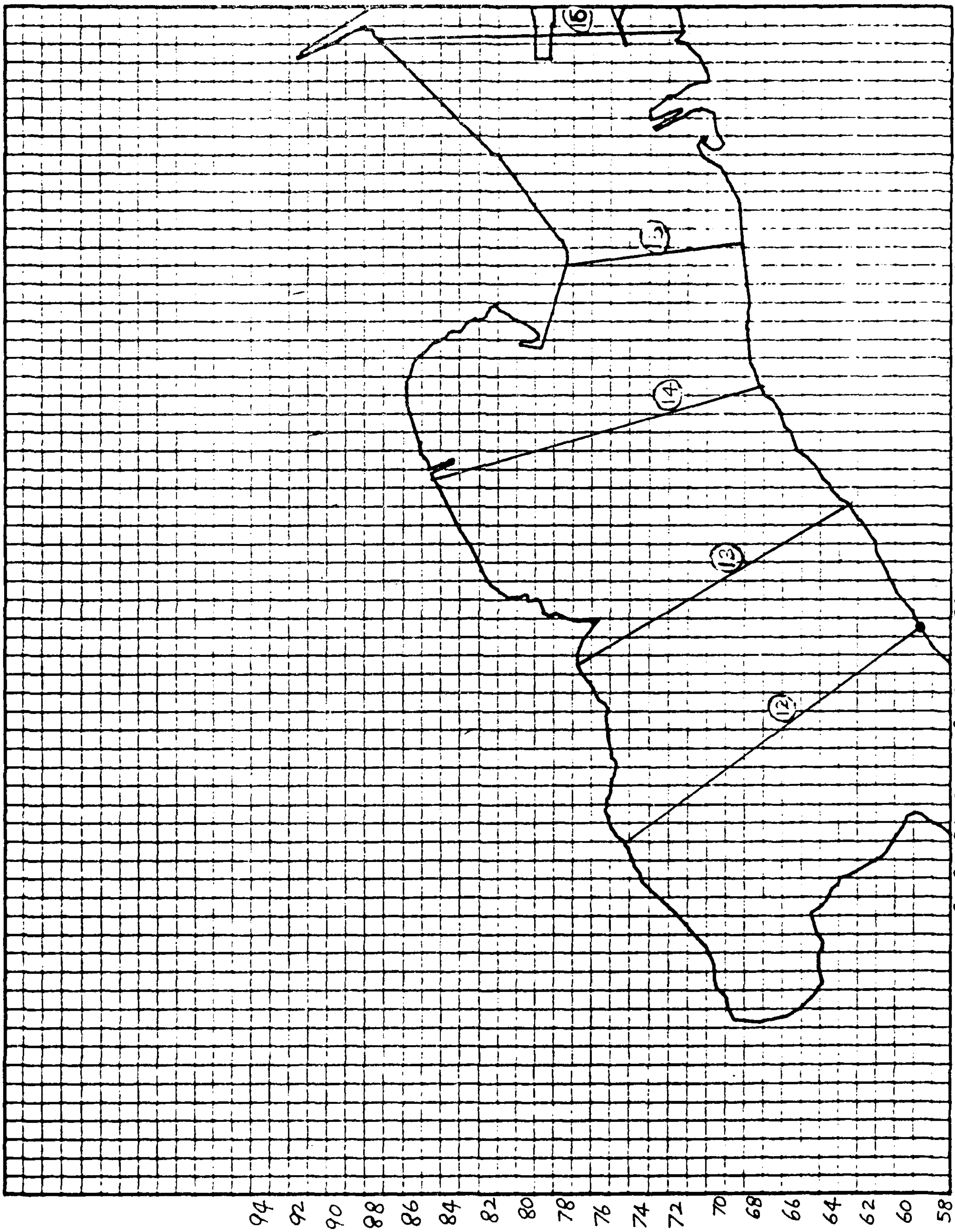
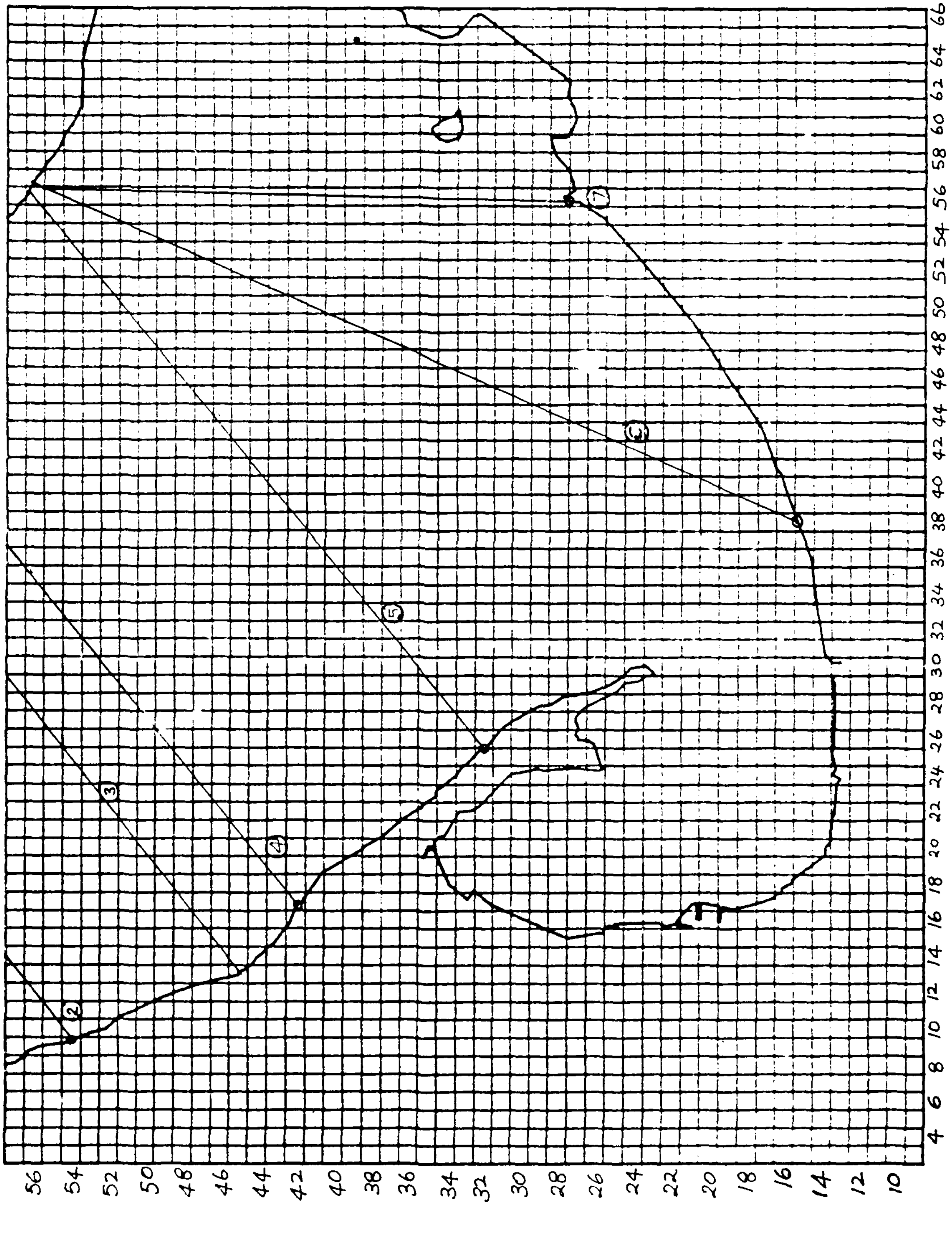
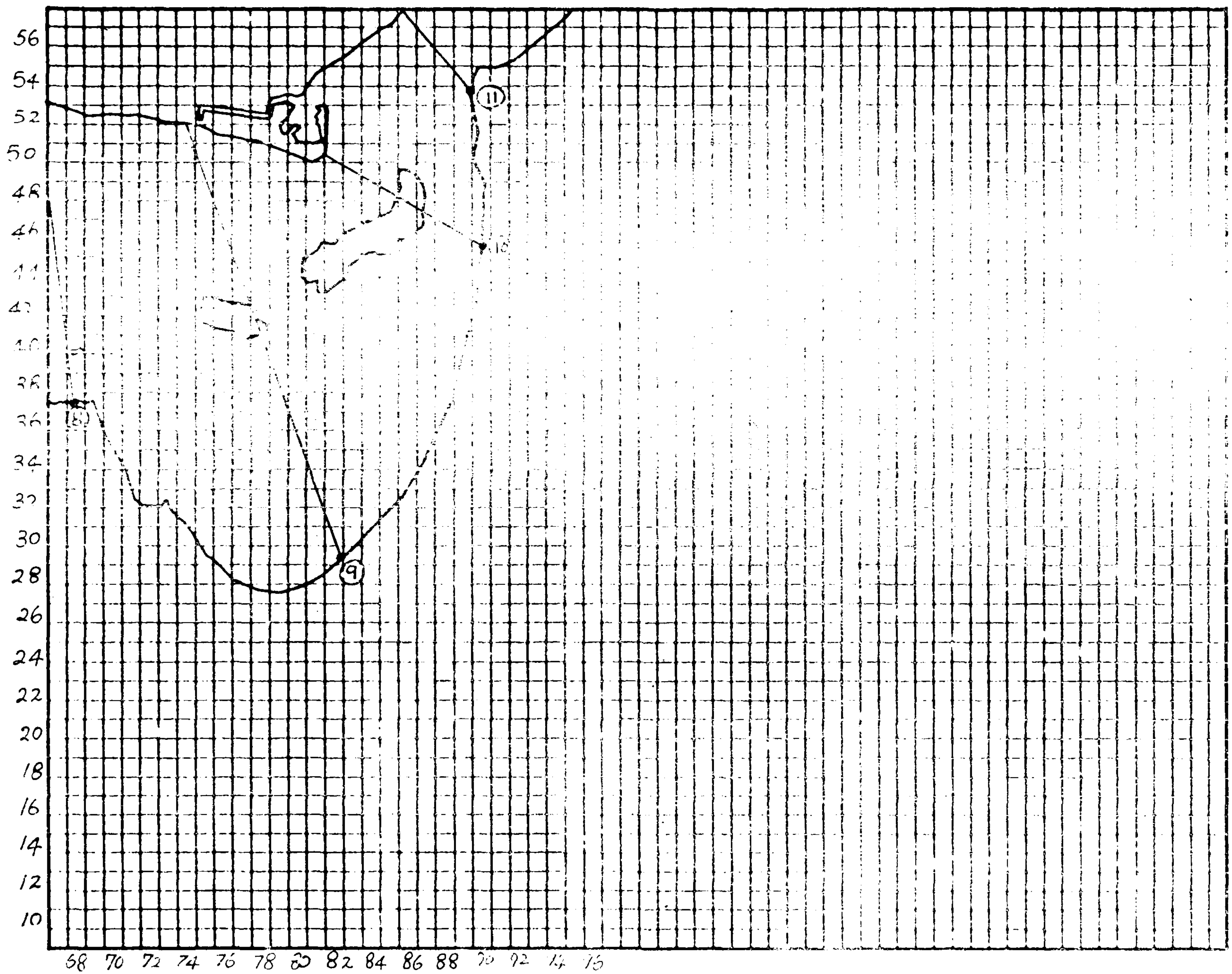

16
14
13
12
68 70 72 74 76 78 80 82 84 86 88 90 92 94 96 98 100 104 108 112 116 120 124 128
94 92 90 88 86 84 82 80 78 76 74 72 70 68 66 64 62 60 58

Chart USM2



4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66
56 54 52 50 48 46 44 42 40 38 36 34 32 30 28 26 24 22 20 18 16 14 12 10
2
3
4
5
7

Chart USM3



56
54
52
50
48
34
30
28
26
24
22
20
18
16
14
12
10
68
70
72
74
76
78
80
82
84
86
88
90
92
11
9

Cha 'SM4